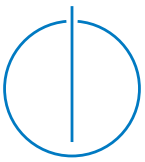
DEPARTMENT OF INFORMATICS

TECHNISCHE UNIVERSITÄT MÜNCHEN

Master's Thesis in Robotics, Cognition, Intelligence

Development and Validation of an Efficient Thruster-Based Control Strategy for Orbital Robots

Florian Schmucker



DEPARTMENT OF INFORMATICS

TECHNISCHE UNIVERSITÄT MÜNCHEN

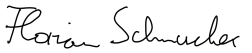
Master's Thesis in Robotics, Cognition, Intelligence

Development and Validation of an Efficient Thruster-Based Control Strategy for Orbital Robots

Entwicklung und Validierung einer effizienten triebwerkbasierten Regelstrategie für Weltraumroboter

Author:	Florian Schmucker
Supervisor:	Prof. Dr. Alin Albu-Schäffer
Advisor:	Davide Calzolari
Submission Date:	July 15, 2020

I confirm that this master's thesis is my own work and I have documented all sources and material used.



Munich, July 15, 2020 Florian Schmucker

Acknowledgments

First and foremost, I would like to thank Alessandro Massimo Giordano for providing guidance and feedback throughout the whole creation process of this work. The many stimulating meetings and conversations were vital in helping me to develop my ideas and have contributed greatly to the improvement of the thesis.

I would like to express my gratitude to my supervisor Prof. Alin Albu-Schäffer for the opportunity to work at the DLR Institute of Robotics and Mechatronics. My sincere thanks go also to my supervisor Davide Calzolari.

Appreciation is due to my student colleagues which provided not only a great working environment but also a made my time more fun.

Lastly, I must express my very profound gratitude to my brother who gave great suggestions during the proof-reading process.

Thank you.

Abstract

In this work it will be shown how the different aspects of the thrusters used in space robots for On-Orbit Servicing can be separately investigated. Not only a closed loop observation of the complete system should be performed, but the thruster actuation of six degrees of freedom wrenches itself can be divided into self-sufficient sub-parts. The goal is to apply the gained insights on the complex system in order to be able to more easily tune the control so that its performance matches the mission specifications.

First, the concept of using thrusters as an actuator with a wrench interface is laid out. The high number of mounted thrusters and their special properties make their utilization non-trivial. Different assignment and modulation methods have to be selected depending on the available computational resources and the requirements on fuel efficiency and accuracy.

A well designed thruster configuration lays out the foundation for the thruster management function to actuate fuel-efficient wrenches. The configuration defines the controllability envelope and thus the maximal solution space of feasible assignments. With the help of the analogy to grasp planning and general results from the theory of positive spanning spaces, it is investigated how a beneficial placement of thrusters could look like. Redundant and decoupled thruster positioning provides the best properties from a performance point of view but requires a high number of thrusters.

The inherent properties of the thrusters lead to a control behavior, which is hard to specify for the complex system but can be easily analyzed for a simplified environment. For very simple controllers, this work develops how the gains can be optimally tuned in order to avoid or at least shape unwanted closed loop behavior.

All previous results are fused and examined in the environment of a realistic space robot simulator for an exemplary control task using a momentum-based controller. For the complex system, simple closed loop characteristics from the one-dimensional task, like limit cycles, extend to more general representations.

Lastly, also the complete system design is put into relation to limitations coming from uncertainties in the configuration and state measurement. Robust observer models are required in order to guarantee a required performance.

Contents

List of Abbreviations

ADR	Active Debrise Removal
CC	Control Capability
COM	Center of Mass
DoF	Degree of Freedom
ENVISAT	Environmental Satellite
ESA	European Space Agency
EVA	Extra Vehicular Activity
IVA	Intra Vehicular Activity
LEO	Low Earth Orbit
LP	Linear Programming
LWR	Light Weight Robot
MF	Modulation Factor
MIB	Minimal Impulse Bit
MILP	Mixed Integer Linear Programming
minCC	Minimal Control Capability
OOS	On-Orbit Servicing
PI	Pseudoinverse
POPC	Passivity Observer and Passivity Controller
PWM	Pulse-Width Modulation
PWPF	Pulse-Width Pulse-Frequency
QP	Quadratic Programming
RCT	Reaction Control Thruster
SDM	Sigma-Delta Modulator
SP	Specific Impulse
STD	Standard Deviation
SVD	Singular Value Decomposition
TMF	Thruster Management Function

1 Introduction

Robotic systems, with varying degrees of autonomy, are becoming increasingly important for different space operations. These systems are not only employed in planetary surface exploration in the form of rovers, but also find usage in a number of Intra Vehicular Activities (IVA) or Extra Vehicular Activities (EVA) [1].

Especially with respect to EVA operations, the growing number of satellites and other orbital structures in space lead to an increased demand for systems capable of helping in construction, maintenance, docking, re-fueling or inspection [2]. In the last decade on average 100 satellites were launched every year [3]. In parallel the number of on-orbit failures has also been exceeding the number of launch failures in recent years and are, therefore, now the most dominant cause of failure. Even when being functional, satellites have to be decommissioned once they run out of fuel. Immediately, the potential of re-fueling but also the controlled de-orbiting in the form of Active Debrise Removal (ADR) come to mind.

This set of problems can be grouped under the term On-Orbit Servicing (OOS). Only some of them can be performed by astronauts and are, in addition, dangerous and subject to limitations [2]. Several studies (e.g. [4]) have demonstrated that robotic OOS can not only be commercially feasible, but will become essential for the future. It will not be sufficient to automatically lift satellites at their end of life from Low Earth Orbit (LEO) into a graveyard orbit, but OOS strategies will become mandatory [5].

1.1 Space Robots for On-Orbit Servicing

A typical space robot forms a servicing system of two components, a base spacecraft and a n-Degree of Freedom (DoF) robot manipulator. Within the scope of the COMRADE project commissioned by the European Space Agency (ESA) the e.Deorbit mission has the goal to capture and de-orbit an Environmental Satellite (ENVISAT). Figure 1.1 illustrates the general concept. A defective target satellite has to be captured by the robot arm and can then be manipulated with the help of the servicing base.

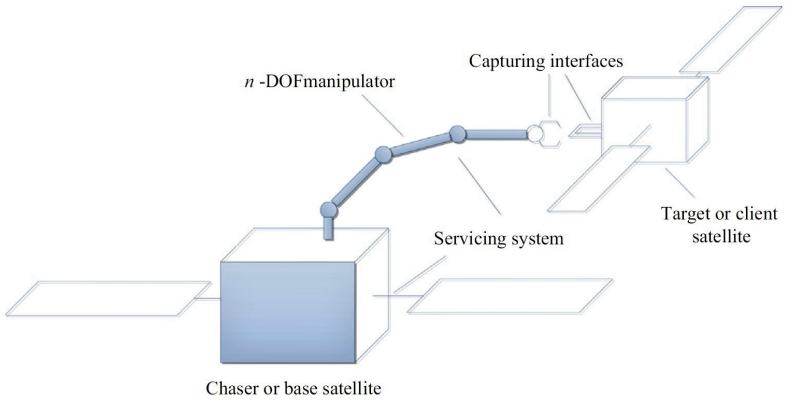


Figure 1.1: Block diagram of a space robot together with a target (retrieved from [3])

The process of capturing a tumbling satellite can be divided into different phases [6]. First, suitable position, velocity and angular rate conditions are induced with far-range rendezvous operations, so that close-range rendezvous operations can be initialized. The main points of this phase lie within the acquisition of the target orbit, reducing the approach velocity and the synchronization between the two objects.

Secondly, during the close-range maneuvers the final approach corridor is opened and afterwards, the mating conditions are ensured by so-called *closing*.

Finally, the mating phase, also called *docking*, can be started. Before the final approach can be conducted, first the physical properties of the satellite are observed and the acquiring motion is planned, which includes the question of how the satellite should be grasped. Afterwards, this is realized during the actual grasping task. In a post-capturing process the target state is stabilized.

Due to the complex dynamics of both the base and the manipulator, the control of the described space robots proves to be difficult. In contrast to standard terrestrial robotics on earth, the lack of gravity, the non-neglectable coupling and the non-fixated base complicate the application of standard control schemes.

Thus, different specialized strategies have been developed and can be grouped into three major categories (see [7] or [8]).

The first category, also often referred to as the *free-flying manipulator* control scheme, is based on an approach that was already proposed in the early 90s [9]. Not only the manipulator, but also both position and attitude of the base spacecraft are actively controlled. Thereby, a manipulator with a theoretically unlimited work-space is obtained. However, due to the coupling of the two subparts, a motion of the manipulator can disturb the pose of the base. As the base is actuated by reaction jets, the counter-actuation leads to an augmented consumption of non-renewable fuel. This can severely limit the lifetime of the system.

In contrast, neither attitude nor position of the base are actuated in the *free-floating* system. Thus, this control scheme saves fuel compared to its previously described counter-part, as only the electric motors of the manipulator are used. Electric energy can always be retrieved through the mounted solar panels. The spacecraft's base is allowed to freely rotate and translate in space. However, the presence of external forces and torques prove to be problematic, as they lead to a drift of the system. If kept unchecked, the system quickly reaches the boundaries of the workspace. Thus, momentum dumping maneuvers have to be employed in order to zero both the linear and angular momentum. As long as any residual momentum is remaining, the drift continues [10]. Another emerging problem is the presence of dynamic singularities, which occur due to the coupling of the structures and are dynamic properties.

Thirdly, as a mix between the previous two categories, only the attitude of the base can be controlled. This leaves out only uncontrolled translational movement.

In order to realize the control outputs, a heterogeneous set of actuators is available [11]. The drives of the robot manipulator and momentum exchange devices, like magnetic torquers and reaction wheels provide internal torques with high rates. Additionally, they only require electric energy, which can be recovered from the solar panels. However, no external forces and torques can be generated by them, which are needed to control the inertial motion of the system. The robot arm can also apply higher torques, than the ones which can be generated by the provided momentum exchange devices.

Therefore, the the additional usage of multiple reaction thrusters is required. Even though they are capable of generating relatively high external torques and forces, they consume fuel in order to do so. Fuel is only available in limited amounts and thus has to be used carefully. The thrusters are also non-linear in their actuation and have a low actuation rate, which further complicates their usage. These characteristics prevent the control wrench to be directly applied to the system dynamics, but first a Thruster Management Function (TMF) has to command the actuation of the thrusters.

1.2 Objective and Outline

Thrusters have a particular set of characteristics, which differentiate them from regular actuators and complicate the control design. In this work, it is investigated how discrete on-off thrusters can be optimally used as an actuator with a given a 6 DoF wrench-interface. A divide and conquer approach is utilized, which allows to generate results on simplified sub-systems. This procedure does not only allow to reduce the number of open parameters in the control of the complete system, but also helps in identifying and minimizing the source of control deviations.

The thruster properties are introduced in Chapter 2, where the dynamics of the spacecraft are disregarded and only the actuation is examined as a separate block, the TMF. This means we can assume to have only a wrench input and have to find a matching thruster actuation using a given thruster configuration. In order to map the desired wrench to an actuation level of the different thrusters, a set of actions is required, which in addition respect the constraints given by the thruster properties. The goal is to make the actuation more optimal in the sense of efficiency and accuracy. As part of this chapter, different state of the art methods to achieve this are presented and expanded by newly proposed adaptions. The chapter finishes with a comparison of the most promising realizations by using a set of performance parameters.

The geometrical distribution of the thrusters on the spacecraft has a large impact on the resulting performance of the thruster actuated system, which is why in Chapter 3 design recommendations of the number, positioning and orientation are given. The shape of the wrench controllability envelope and the minimal fuel consumption within the envelope are directly dependent on these parameters. Convex analysis and the analogy of a robotic grasping problem are used to optimize the configuration.

In contrast, Chapter 4 uses a simplified one-dimensional thruster configuration together with simplistic control types, namely PD- and D-control, in order to investigate how the thrusters behave in a closed loop environment. The focus in this chapter lies on the question of how the control parameters can be adjusted in order to shape potentially occurring limit cycles and other control behavior stemming from the actuation with thrusters. Relations are developed, which link the control performance directly to both the thruster characteristics and the selected thruster assignment methods.

The results of the previous chapters are then merged in Chapter 5, where a complex control task is first introduced and then realized with the help of the thruster actuation. It is investigated how the real space robot is capable of performing the maneuvers required for docking in OOS when using a newly developed assignment algorithm. A realistic simulation environment and a novel control approach is used here.

Finally, Section 5.4 provides a small impression on how different uncertainties in both the spacecraft's structure and the control affect the control performance. The effect of a number of selected uncertainties is analyzed using a Monte Carlo like sampling. Conclusions are drawn with respect to the design of the thruster actuation system.

2

Thruster Actuation

This chapter mainly focuses on the question of how the different parts of the TMF, as visualized by Figure 2.1, can be realized in detail. This means that the TMF is regarded as a self-contained block in this chapter. We want to analyze the specific input-output behavior. Other previously mentioned and commonly used actuators, like reaction wheels or the joint motors, are either locked or decoupled from the control wrench and the dynamic properties are not important. Solely the part of the control wrench which has to be actuated by the thrusters is considered. Technically, the inverted wrench mapping is not part of the TMF, which runs onboard, but only a part of the dynamics model. This wrench input is time-discretely updated depending on the control time interval t_{ctrl}. The output wrench $\tilde{\mathbf{T}}^B$ is also changing time-discretely.

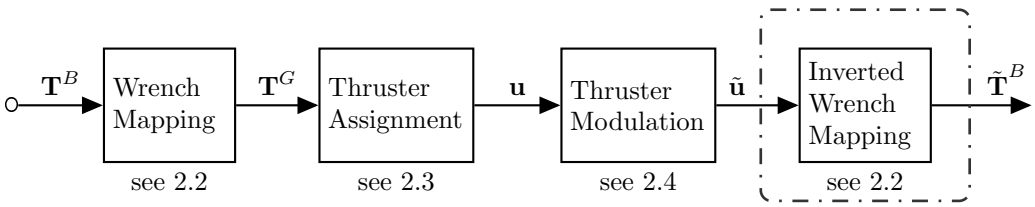


Figure 2.1: Block diagram of the thruster management function

The performance can be determined by the difference of in- and output, the required fuel and additional values we will define later. We do not need any further knowledge from outside, with the exception of some parameters which will also be discussed during the course of this chapter.

We divide the actuation into four different phases.

- The knowledge of how the thrusters are mounted on the satellite body is usually given in a special coordinate frame, the geometrical frame. In contrast, the control wrench is generally computed with respect to the body frame. Thus, in the first phase the relative frame differences have to be used in order to map a control wrench from the body to the geometrical frame.

- A demanded wrench can often be actuated by selection of multiple combinations of thrusters. In this step it is often assumed that the thrusters can apply continuous thrust, which is only upper bounded. It is not straightforward to identify the most optimal assignment of firing thrusters due to redundancies in the configuration. The continuous assignments have to be identified explicitly.
- The discrete nature of the thrusters requires the TMF to generate pulses, which modulate the continuous assignments and discretize them in a sequence of on-off thrusts over time.
- Lastly, the applied wrench can then be transferred back into the body frame in order to update the system states. In this step we can also model uncertainties of the thrusters. This is not part of the TMF, but only required for the system model.

An oversight of the thruster characteristics can lead to a deviation in the actuated wrench compared to the demanded input wrench, which comes from the controller. Additionally, non-idealities like misalignments can negatively affect the performance. Each phase will be discussed in more depth in their respective section.

For the example of an attitude determination and control system for satellites, [12] provides a general overview of actuation approaches.

Note that the actuation can also be directly embedded into the control algorithms, which are often based on Lyapunov stability theory (e.g. [13]). Moreover, these control algorithms are similar to sliding mode controls.

With the emergence of neural networks, these are started to be used for space control applications as well. For example, [14] uses a neural network based approach for attitude control with thrusters.

However, this thesis will solely focus on the previously sketched architecture.

2.1 Thruster Properties

The used one-sided thrusters, in the literature also referred to as Reaction Control Thrusters (RCT), have several properties that distinguish them from an ideal actuator. The main goal is to use the thrusters in a manner that minimizes both the in-output error, from now on also referred to as *wrench error*, and the required effort. Ideally, both properties are independent from the control frequency f_{ctrl}, which defines the update rate of the input. With some assignment and signal modulation approaches, it is possible to directly embed the most important properties. Every unmodeled property will increase the wrench error applied during each control time interval.

2.1.1 On-Off Nature

Proportional thrusters, as the name indicates, open the fuel valves proportional to the commanded thrust. However, these are rarely used in space robots mainly due to three reasons.

First, they are very complex and difficult to build. Second, they tend to have a high hysteresis. Third, in order to reduce the amount of thrust, the valves have to be adjusted proportionally. A very small thrust is realized by a small opening of the valve. In the extreme conditions present in space, it is easily possible that the valves then cannot fully close anymore due to the accumulation of small particles [15].

These reasons are why, for spacecraft applications, thrusters are only used in on-off mode. They either apply zero thrust or the full nominal thrust value u_{max}. This kind of thruster resolves the issue of jammed valves and consequently minimizes fuel leakage, but on the downside also complicates the TMF.

One of the main tasks of the TMF becomes apparent. The continuous-valued control wrenches have to be translated to a matching sequence of thrust pulses.

2.1.2 Minimal Valve Opening Time

Additionally, the valves have to be open for at least a small amount of time, usually several milliseconds. As a result the pulse sequences have a lower limit. A pulse of one thruster cannot be shorter than this specified time frame, but can always be longer. In this context, the literature often uses the term of Minimal Impulse Bit (MIB), which is why this constant will be referred to as t_{mib} in this work.

The minimal thruster firing time heavily influences the performance of the TMF. During the signal modulation it has to be verified that no pulses shorter than t_{mib} are commanded. Translated into thrust space, where we are not looking at firing times but at thrust assignments u_i, this means that assignments $u_i < u_{dead}$ for a thruster with index i cannot be carried out. Note that u_{dead} is also depending on the signal modulation and has only a direct meaning in the context of a specific one. Currently, u_{dead} is only an abstract concept. Two approaches can be considered for the question of how to handle this property. The assignment step can incorporate this parameter so that the thrusters have only assignments from the set $u_i \in \{0; u_{dead}, ..., u_{max}\}$. Alternatively, any assignment $u_i < u_{dead}$ is mapped to zero before the signal modulation. It gets apparent why we denoted this parameter with u_{dead}. Any assignment below this value will not be actuated and results in a dead assignment. Obviously, such a correction will lead to an unexpected wrench error.

2.1.3 Limited Number of Switching Operations

Any of the integrated valves is only guaranteed to work for a limited amount of operations, which is defined by the manufacturer. By exceeding this number, the valve is more prone to break, which makes the usage of some thrusters impossible and therefore might make the whole system unusable. Even though the number of operation cycles is in the area of hundreds of thousand, which automatically results in a large margin, it prolongs the serviceable life-time of the space robot if this property is explicitly considered.

Due to this reasoning, there already exist some approaches that include it in the thruster assignment problem [16].

Though, usually the number of switching operations is not explicitly considered in general assignment formulations. It is implicitly assumed that the number is sufficiently high for the fulfillment of the mission.

2.1.4 Bounded Controllability Envelope

The solution space of wrenches $\boldsymbol{T}$ which lead to feasible assignments, is inherently bounded due to the used thruster configuration and the previously mentioned thruster properties t_{mib} and u_{max}. Considering an ideal signal modulation, we can ignore the on-off nature and imagine a continuous six-dimensional volume. The boundaries of this volume cannot be easily calculated, as the dimensions are dependent on each other.

However, we can lock up the real envelope by a six-dimensional hyper-cube. The cube can be derived by examining each dimension separately and by taking the maximally and minimally possible force or torque value. The real envelope containing all properties is only a subset of the theoretical hyper-cube and is generated by a complex function, which is not directly computable. Depending on the thruster assignment, the obtainable envelope can also be an even more reduced version of the theoretical one.

The thruster assignment has to map any demanded control wrench outside of the set $\boldsymbol{T}$ to the closest one inside. Generally speaking we map onto the boundary of the volume to actuate a control vector, which ideally minimizes the weighted squared error. The control vector should be only scaled by length, but keep its pointing direction. This also means that control forces and torques should not change their sign.

This operation has to be done in particular when the control vector gets too long.

2.1.5 Disregarded Thruster Properties

There still exist a number of properties of both the thrusters and the propulsion system that were and will not be included within the scope of this work.

Degrading Thrust and Number of Valves In order to save costs and space, the number of fuel tanks is usually lower than the number of thrusters. One tank is connected to multiple pressure valves. The pressure and thus also the applied thrust then falls of non-linearly depending on the number of simultaneously actuated thrusters using the same tank. This means the performance degrades and the control wrench is only actuated with a large error. The Specific Impulse (SP) measures how effectively the propellant is used.

Additionally, a further limitation can take place if also not each thruster has its own valve. This then results in limitations on the possible combinations of on-thrusters.

Rise and Fall Time In this work we assume that the switching of thruster states happens instantaneously. The time behavior of the applied thrust is assumed to follow an ideal step behavior.

Realistically however, when switching a thruster from off- to on-state, the full thrust is approximately applied only after a specific start time T_s. Analogously, the shutdown time T_{sd} specifies the transition time from on- to off-state (see Chapter C.2 in [17]). Often it is assumed that both dynamic time properties are fast compared to the general switching time and thus can be ignored.

Non-Idealities During the thruster assignment we not only ignore the rise and fall time of the applied thrust, but also assume perfect thrusters, which ideally follow their nominal specification. In reality, each thruster has a bias with respect to its nominal thrust value. Also, the repeatability of each off-on-off cycle is not perfectly given. Therefore, the thrust additionally jitters around its bias value for each thruster. This property is often called roughness in the literature.

For now it is assumed that the mounting of the thrusters can be done without uncertainties. Elevation and azimuth angles as well as the position of the thrusters with respect to the body frame are perfectly aligned.

During any maneuver it is possible that the movement of the fuel inside the tanks can interact with the complete system (see [15], Chapter 9). The consumption of fuel also reduces the mass of the space robot during operation. Both properties can lead to uncertainties in the dynamic models of the spacecraft.

In Section 5.4 the effect of some of these non-idealities will be examined in a closed control loop. More precisely, the focus will be on answering the question of the influence of different effects in the wrench error.

2.2 Grouping of Thrusters

In order to achieve given 6 DoF wrench tasks, not only one but multiple thrusters have to be used. For an arbitrary number of n thrusters, they can be grouped into a thruster mapping matrix $\mathbf{B}$ depending on their geometrical position and orientation, as shown in Subsection 2.2.1. An explanation of how wrenches can be mapped between different frames will be part of Subsection 2.2.2.

2.2.1 Thruster Configuration

At first we have to take a look into the geometrical structure of the thruster mounting. Once specified, the mounting is fixed for the entire operation time of the space robot.

Each thruster can only apply a non-negative thrust u_i along its pointing direction, which is the reverse of its firing direction. Often this property is also denoted as *one-sided*. We will only focus on these thrusters in this work. For a thruster i, the resulting wrench $\boldsymbol{T}_i^G$, given in the geometrical frame, can be clearly identified by its location and orientation. Only five parameters need to be specified due to symmetry of rotation along the pointing axis.

The orientation is specified by azimuth $\chi_i^G \in]-\pi, \pi]$ and elevation angle $\gamma_i^G \in [-\frac{\pi}{2}, \frac{\pi}{2}]$. Both together define the axis along which the thruster applies its thrust u_i. The resulting force vector $\mathbf{F}_i^G = \begin{bmatrix} F_{i,x}^G & F_{i,y}^G & F_{i,z}^G \end{bmatrix}^T$ can then be calculated with

$$\begin{bmatrix} F_{i,x}^G \\ F_{i,y}^G \\ F_{i,z}^G \end{bmatrix} = \begin{bmatrix} \cos(\chi_i) \cdot \cos(\gamma_i) \\ \sin(\chi_i) \cdot \cos(\gamma_i) \\ \sin(\gamma_i) \end{bmatrix} \cdot u_i \tag{2.1}$$

Figure 2.2 visualizes the previous definition. The azimuth angle is defined in the x-y-plane starting from the x-axis. A positive elevation angle leads to a positive force in z-direction. The red vector depicts the pointing direction of a hypothetical thruster. Note once again that the angle definition of the pointing axis is the opposite of the firing direction of a thruster.

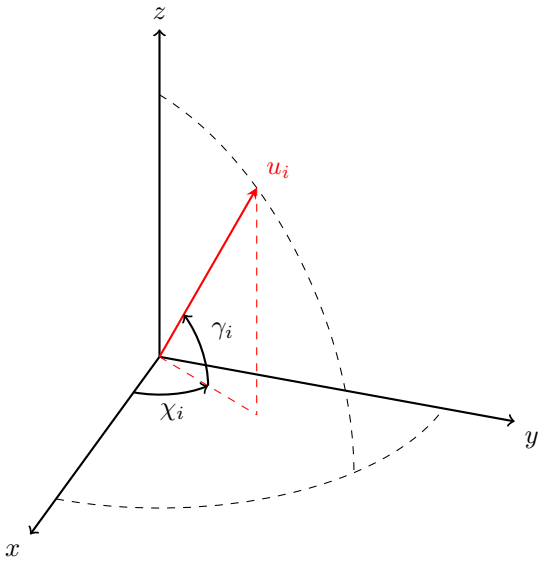


Figure 2.2: Definition of the thruster orientation

The position is provided by the displacement $\mathbf{r}_i^G = \begin{bmatrix} r_{i,x}^G & r_{i,y}^G & r_{i,z}^G \end{bmatrix}^T$ of each thruster from the frame origin. The cross product of displacement and Cartesian forces produces the respective torques $\boldsymbol{\tau}_i^G = \begin{bmatrix} \tau_{i,x}^G & \tau_{i,y}^G & \tau_{i,z}^G \end{bmatrix}^T$

$$\boldsymbol{\tau}_i^G = \boldsymbol{r}_i^G \times \boldsymbol{F}_i^G = \begin{bmatrix} r_{i,y}^G \sin(\gamma_i) - r_{i,z}^G \sin(\chi_i) \cos(\gamma_i) \\ r_{i,z}^G \cos(\chi_i) \cos(\gamma_i) - r_{i,x}^G \sin(\gamma_i) \\ r_{i,x}^G \sin(\chi_i) \cos(\gamma_i) - r_{i,y}^G \cos(\chi_i) \cos(\gamma_i) \end{bmatrix} \cdot u_i \tag{2.2}$$

It gets apparent how the knowledge about the geometry can be used to construct a thruster mapping matrix $\mathbf{B} \in \mathbb{R}^{6xn}$, which directly maps the applied thrusts of n thrusters to the corresponding wrench T^G in the geometrical frame.

$$\mathbf{T}^G = \begin{bmatrix} \boldsymbol{F}^G \\ \boldsymbol{\tau}^G \end{bmatrix} = \mathbf{B} \cdot \mathbf{u} \tag{2.3}$$

with $\mathbf{u} = [u_1\ u_2\ ...\ u_n]^T$ being the stacked vector of the n thrusts. Each column $\mathbf{b}_i$ in $\mathbf{B}$ is constructed from the orientation and position of the respective thruster i. The first three rows describe the resulting forces and the last three rows the torques.

2.2.2 Mapping to Geometrical Thruster Frame

Note that at this point we can only calculate the applied wrench in the geometrical frame of the thrusters. However, it is generally desired to express it in the body frame. In general, it cannot be assumed that both frames are identical.

The matrix $\boldsymbol{R}_B^G$ rotates a vector from the body frame B to the geometrical frame G. Additionally, the vector $\boldsymbol{p}_B^G$ indicates the positional displacement of the body frame as viewed from of the geometrical frame. With both parameters given and under the assumption that both are constant with respect to time, the mapped wrench can be given by

$$\boldsymbol{T}^B = \begin{bmatrix} \boldsymbol{R}_G^B & \mathbf{0} \\ -\boldsymbol{R}_G^B \hat{\boldsymbol{p}}_B^G & \boldsymbol{R}_G^B \end{bmatrix} \boldsymbol{T}^G = \left(\boldsymbol{Ad}_B^G\right)^T \boldsymbol{T}^G \tag{2.4}$$

with $\hat{\boldsymbol{p}}_G^B$ being the skew-symmetric cross-product matrix. The matrix $\boldsymbol{Ad}_B^G$ is also called adjoint transformation matrix and can be used to map wrenches $\mathbf{T}$ and twists $\boldsymbol{\xi}_B^G$ from one frame to another.

An in-depth derivation, using the notion of dual variables and power ports, can be found in the appendix in Chapter A.

In an analogous fashion we can calculate the mapping in the opposite direction by inverting $\left(\boldsymbol{Ad}_B^G\right)^T$. This can be always done due to the properties of the adjoint matrix.

2.3 Thruster Assignment Problem

The previous section explained how we can map from thrust to wrench space. However, for a control task it is necessary to find a mapping in the opposite direction, which is more difficult to obtain.

For most spacecraft applications the configuration is chosen to be redundant. The number of thrusters exceeds the number of dimensions in the wrench vector $\mathbf{T}^G \in \mathbb{R}^6$. Therefore, we cannot easily invert $\mathbf{B}$ in order to get a unique solution of the appropriate mapping. The system is underdetermined so that in theory an infinite amount of solutions might exist. More precisely, we can get zero or up to an infinite amount of possible continuous assignments of thrusters which perfectly satisfy equation 2.3.

We need to add different constraints in order to rank all possible assignments and only select the best assignment.

This results in the problem of finding a suitable thruster assignment for a given control wrench $\mathbf{T}^B$. The on-off nature is ignored during this step, as it is assumed that the signal modulation is responsible for the conversion of the continuous assignments to thruster pulses. Thus, the task of the thruster assignment mainly focuses on the question on how to select the best assignment from the number of possibilities.

First, the assignment should result only in a minimal wrench error by respecting as much thruster properties as possible. By ignoring important ones like the minimal valve opening time, the signal modulation might enforce some adjustments in the assigned values and thus introduces an unmodeled wrench error with its origin in the assignment phase.

Secondly, the fuel consumption has to be minimized, often indicated by the sum of all assigned values u_i in each thruster.

Thirdly, different other parameters should be considered implicitly or explicitly. The number of on-thrusters per time should be minimal in order to increase the life time of the system and also implicitly decrease the effect of degrading thrusts. Ideally, rarely used valves and thrusters should be preferred during the thruster selection. The goal here is to homogenize the wear-down of both valves and thrusters in order to reduce the risk of losing some controllability due to break-down of single thrusters early in the system's lifetime.

Lastly, it has to be also considered that the perfect solution might be difficult to obtain in real-time in an online manner or is even computational infeasible to calculate. Thus, the computational load has to be kept in mind.

The thruster assignment approach used in the end can only be a trade-off between these different parameters. For example, a more accurate solution with less wrench error might need significantly more fuel and a higher number of thrusters.

In the literature, various approaches to solve the allocation problem have been proposed. Generally one can identify three different main categories. First, we have direct methods that try to solve equation 2.3 either directly or by inverting $\mathbf{B}$, resulting in a pseudoinverse, which can be further optimized.

The second category unites different constrained optimization methods. This often results in Linear Programming (LP) problems, solved mainly by the *Simplex* or the *Primal-Dual Interior Point* algorithm. More complex adaptions will be proposed during this work. This category also includes allocation algorithms, which are based on subgradient optimization [18], which however will not be investigated further in the following.

Historically, due to the limited computational power available in satellites, pre-computed tables were often used and form the third group. Online, only the best thruster assignment from the set of pre-computed ones has to be selected for a given control vector.

In the following subsections different meaningful approaches will be presented and later also compared against each other. Each method has a different set of strengths and weaknesses and the finally chosen approach can only be a trade-off with respect to the previously presented performance parameters.

In [19], a similar overview, as given in this work, can be found, which also focuses on the question of how to optimally derive the LP problem formulations.

There exist also different kinds of categorizations. For example, the thruster assignment problem can be only divided into fixed and flexible approaches [20].

2.3.1 Pseudoinverse Solution

An obvious solution to the assignment problem is the use of the Moore-Penrose Pseudoinverse (PI) $\mathbf{B}^\dagger$ in order to get

$$\mathbf{u} = \mathbf{B}^\dagger \cdot \mathbf{T}^G \tag{2.5}$$

which allows to automatically calculate assignments from any given control vector in the geometrical frame. This inverts equation 2.3.

$\mathbf{B}^\dagger$ can be derived by calculating

$$\mathbf{B}^\dagger = \mathbf{B}^T \left(\mathbf{B}\mathbf{B}^T + \epsilon \mathbf{I} \right)^{-1} \tag{2.6}$$

with $\mathbf{B}^T$ indicating the transpose of $\mathbf{B}$. Note that depending on the thruster mounting in theory it is possible that $\mathbf{B}\mathbf{B}^T$ becomes singular. Thus, it is necessary to introduce an identity matrix, scaled by a very small scalar value ϵ, that guarantees a possible inversion. However, as we will see later for the space robot, we will only use configurations which have full row rank. As a result we can set $\epsilon = 0$ without any problem.

This kind of PI minimizes the L2-Norm of the assignment vector $\mathbf{u}$. Otherwise, no constraints are set regarding possible solutions. In comparison to other norms, generally a lot of thrusters will be engaged in the assignments, as multiple small assignments are preferred to a sparse assignment vector.

The resulting assignments may also exceed the nominal thrust or can alternatively be even negative. Negative thrusts are not possible due to the one-sided thrusters, which can only fire in one direction. This requires to cap the assignments before the signal modulation so that all values for the n thrusters satisfy the inequality $0 \leq u_i \leq u_{max}$. Independent from the selected signal modulation also not arbitrary small assignments can be carried out due to t_{mib}. This property is not included either. After all these corrections, this may result in a very inaccurate mapping.

We can use the notion of the null-space in order to improve this first assignment. We can dissect the mapping matrix $\mathbf{B}$ with a Singular Value Decomposition (SVD)

$$\mathbf{B} = \mathbf{U}\mathbf{S}\mathbf{V}^T = \mathbf{U}\left[\mathbf{S}_\sigma\, \mathbf{S}_{null}\right]\left[\mathbf{V}_\sigma\, \mathbf{V}_{null}\right]^T \tag{2.7}$$

$\mathbf{S}_\sigma$ is a positive definite diagonal matrix $\in \mathbb{R}^{6x6}$. The diagonal entries are equal to the singular values of $\mathbf{B}$. $\mathbf{S}_{null} \in \mathbb{R}^{6x(n-6)}$ is a zero matrix. The matrices $\mathbf{U} \in \mathbb{R}^{6x6}$ and $\mathbf{V} \in \mathbb{R}^{nxn}$ have orthonormal columns, while the ones of $\mathbf{V}_{null} \in \mathbb{R}^{nx(n-6)}$ additionally span the null space of $\mathbf{B}$.

We can now formulate an identical formulation for the Moore-Penrose inverse using the SVD

$$\mathbf{B}^\dagger = \mathbf{V}_\sigma \mathbf{S}_\sigma^{-1} \mathbf{U}^T \tag{2.8}$$

Due to the redundancy of the thruster configuration, we have a high-dimensional null space of size $\mathbb{R}^{nx(n-6)}$. It can be used to get a better assignment, which satisfies more of the thruster properties. This reduces the wrench error and also leads to improvements in the fuel consumption. We can add a null space assignment $\mathbf{u}_{null}$ to any assignment vector $\mathbf{u}$ without changing the theoretically applied wrench, when using equation 2.3.

$$\mathbf{u} = \mathbf{B}^\dagger \cdot \mathbf{T}^G + \mathbf{u}_{null} = \mathbf{B}^\dagger \cdot \mathbf{T}^G + \mathbf{V}_{null}\, \mathbf{h}_{null} \tag{2.9}$$

$\mathbf{h}_{null}$ is an arbitrary vector $\in \mathbb{R}^{(n-6)}$.

However, an optimal and static selection of $\mathbf{h}_{null}$ or $\mathbf{u}_{null}$ is difficult to make. For any given control wrench the assignment has to be meaningful.

If we are assuming a symmetric thruster configuration one can intuitively improve the assignment quality. Symmetric thruster configurations have the property that any assignment $\mathbf{u} = c \cdot \mathbf{1}$ for an arbitrary coefficient c is in the null space. $\mathbf{1}$ is a vector of ones $\in \mathbb{R}^{(n)}$. This means, we can easily select

$$\mathbf{u}_{null} = |\min(\mathbf{B}^\dagger \cdot \mathbf{T}^G)| \cdot \mathbf{1} = u_{low} \cdot \mathbf{1} \tag{2.10}$$

to translate so that all assignments have a positive value.

This will greatly reduce the wrench error, which results from the clipping of infeasible values. However, only very fuel expensive assignment will be obtained. Negative assignments indicate that the thruster should fire in its opposite intended direction, which however is not possible. If we only translate the assignment as discussed, some small assignments $u_{low} < u_i < 0$ will now have a positive thrust. Only due to the also shifted positive assignments no wrench error is made. However, it is now also more likely that these exceed u_{max} and thus, inadvertently, a rotation of the actuated wrench in the wrong direction might be possible.

If every thruster has an antagonist in the configuration and if the configuration is also symmetric, we can just assign

$$\mathbf{u} = 2 \cdot \mathbf{B}^{\dagger} \cdot \mathbf{T}^{G} \tag{2.11}$$

and accept the clipping of negative values. This results in a relatively good static inverse, which is also relatively fuel conservative. Compared to other PI solutions the number of simultaneously active thrusters is also quite low. An antagonistic configuration is specified by the fact that the applied thrust from one thruster can always be negated by a thrust with the same amplitude of only one other thruster. In such configurations both antagonistic thrusters will be assigned the same thrust value, but with a different sign. By doubling the positive values, we can simply ignore the negative ones without a resulting wrench error.

The computational simplicity and little memory requirements for using a static PI for the assignment problem make it tempting to use it. Though, without a null space optimization, similar to the one presented, the wrench error will be intolerable, especially for large control wrenches. The main problem still lies within the question of how a good $\mathbf{u}_{null}$ can be obtained for arbitrary configurations. There also exist some other attempts to develop methods, which derive a static null-space assignment $\mathbf{u}_{null}$ for arbitrary thruster configurations. Namely [21] and [22] experimented with the introduction of cost-functions, mainly different norms with additional constraints, in order to improve the static PI solution. Their approach resulted in a min-max-optimization that has to be carried out once offline. The adapted PI can be calculated offline and then stored in the memory. Online only a matrix multiplication has to be carried out.

An iterative approach, which uses the so-called *Redistributed Pseudoinverse*, can further reduce the wrench error. It was implemented in [23] and slightly adopted in [24]. The authors developed this method for a different use-case in an aircraft.

The main idea is to first calculate an initial assignment using equation 2.5. If the assignment contains values outside of $0 \leq u_i \leq u_{max}$ for at least one of the n thrusters, the initial mapping matrix $\mathbf{B}$ has to be adapted. The columns of the saturated thrusters are zeroed out and a new pseudoinverse is calculated from there. Additionally, the contributions of the maximally firing thrusters are considered by altering the control wrench accordingly. Afterwards, we calculate a new assignment with equation 2.5 using the altered pseudoinverse and control wrench. This is continued for as long as there are assignments that exceed the boundary values during each iteration. Also, the reduced mapping matrix has to have at least one non-zeroed entry.

If the algorithm terminates, the assignment is a combination of the clipped thrusters and the potentially left-over assignments from the last step before convergence. We can see the pseudo-code for this algorithm in Algorithm 1. Note that in our formulation we only clipped the negative valued thrusters, if no assignments exceeded u_{max}. This potentially reduces the final wrench error, as these thrusters are longer available and might still be used in later iterations.

Algorithm 1 Adapted Iterative Pseudoinverse Algorithm

procedure [ThrustAssign] = IterativePseudo($\mathbf{B}$, $\mathbf{T}_{des}$, u_{max})
  $bConverged \leftarrow false$ ▷ Initialize loop condition
  $\mathbf{B}_{clip} \leftarrow \mathbf{B}$ ▷ Initialize clipping matrix
  $\mathbf{u}_{full} \leftarrow \mathbf{0}$ ▷ Initialize thrust vector with fully firing thrusters

  while bConverged == false **do**
    $\mathbf{B}_{pseudo} \leftarrow$ Moore-Penrose-Inverse of $\mathbf{B}_{clip}$
    $\mathbf{u}_{run} = \mathbf{B}_{pseudo} \cdot \left(\mathbf{T}_{des} - \mathbf{B} \cdot \mathbf{u}_{full}\right)$ ▷ Preliminary pseudo-assignment
    $\mathbf{i}_{max} \leftarrow \mathbf{u}_{run} > u_{max}$ ▷ Flag thrusters which violate upper bound

    Set columns flagged by $\mathbf{i}_{max}$ in $\mathbf{B}_{clip}$ to zero
    Add fully firing thrusters from $\mathbf{i}_{max}$ to $\mathbf{u}_{full}$
    if Sum of $\mathbf{i}_{max} == 0$ **then** ▷ No thruster exceeds umax
      $\mathbf{i}_{min} \leftarrow \mathbf{u}_{run} < 0$ ▷ Flag thrusters which violate lower bound
      Set columns flagged by $\mathbf{i}_{min}$ in $\mathbf{B}_{clip}$ to zero
      if Sum of $\mathbf{i}_{min} == 0$ **then** ▷ No thruster below 0
        $bConverged \leftarrow true$

    if $\mathbf{B}_{clip} == \mathbf{0}$ **then** ▷ Zero matrix
      $bConverged \leftarrow true$

  ThrustAssign $\leftarrow \mathbf{u}_{full} + \mathbf{u}_{run}$ ▷ Final assignment

Similar to the other previously presented PI approaches the minimal valve opening time and a direct optimization of the number of active thrusters still cannot be carried out. Also similarly, the maximal controllability envelope cannot be utilized. However the run-time still is relatively low, as we usually require only a few iterations until we converge to a solution. A quantitative analysis will be carried out in Section 2.5, where the different approaches are also compared against each other.

2.3.2 Vanilla Linear Programming Formulation

The approach of the Redistributed Pseudoinverse does not use a static inversion of the mapping matrix. Obviously, an iteratively adjusted matrix will result in a reduced wrench error across all possible control wrenches. We can go even further and directly formulate the assignment problem as a constrained optimization problem in order to get the solution with the most optimal fuel consumption. Different optimization problems can be formulated depending on the considered properties and design goals.

The most basic approach results in a LP problem. Only the control wrench and the boundedness of each thruster are included as constraints, while the optimization parameter is the fuel consumption.

We want to select the most fuel efficient thruster assignment from the set of all assignments that actuate the control wrench. Once again it has to be stressed that this formulation still ignores the on-off nature of the thrusters. Additionally, the minimal valve opening time is disregarded, so that the assignments are only preliminarily valid if they range from $0 \leq u_i \leq u_{max}$.

This results in the optimization problem

$$\min_{u} \quad \mathbf{k}_1^T \mathbf{u} \tag{2.12a}$$

$$\text{subject to} \quad \mathbf{B} \cdot \mathbf{u} = \mathbf{T}_{des}^G, \tag{2.12b}$$

$$0 \leq u_i \leq u_{max} \quad \text{for } i \in [1...n]. \tag{2.12c}$$

with $\mathbf{k}_1$ being a weighting vector with constants, which are defined so that the inner product with $\mathbf{u}$ is a proper fuel consumption. Often, a vector of ones can be used for $\mathbf{k}_1$, assuming that all thrusters have the same efficiency.

This problem can be solved with any standard linear programming solver using the Simplex or an Interior-Point algorithm. No assignment can be found if the wrench is outside of the controllability envelope, as constraint 2.12b will become infeasible. Note that this formulation can lead to a wrench deviation when applied together with realistic thrusters with a nonzero minimal firing time. Assignments below u_{dead} will lead to no firing during the control time interval. This will occur independent from the selected signal modulation method. From our point of view it is also problematic that for us the wrench error is nondeterministic. For any demanded control wrench, no predictions can be made whether and in which entries a wrench error is made. Let us imagine that only forces are demanded. By accident it is now possible that rotations around the body axes are excited due to a torque contribution.

In order to reduce this wrench error, it is possible to add an operation before the signal modulation. This corrects assignments according to the following rule

$$u_i = 0 \qquad u_i < \frac{u_{dead}}{2} \tag{2.13a}$$

$$u_i = u_{dead} \qquad \frac{u_{dead}}{2} \leq u_i < u_{dead} \tag{2.13b}$$

$$u_i = u_i \qquad u_{dead} \leq u_i \leq u_{max} \tag{2.13c}$$

for $i = [1, ..., n]$. Previously infeasible assignments are rounded towards the next feasible value. Without this adaption, all these assignments would have been set to zero.

Also the number of active thrusters is not explicitly optimized, but it is only implicitly assumed that this property correlates with the fuel optimality.

2.3.3 Mixed Integer Linear Programming Formulation

We now want to fix the issues of the LP formulation by selectively penalizing any assignment that contains a value, which is located inside the dead-zone $0 < u_i < u_{dead}$. As we will still optimize for the fuel consumption, we add a penalty term to the regular fuel consumption. As soon as the dead-zone condition is fulfilled for any thruster, this penalty should be added towards the actual fuel consumption. Logical constraints are required to transfer this demand to an optimization formulation. The idea of transferring some parts of the assignment problem into logical constraints is not entirely new and, for example, has been investigated by [25] for a different task.

Additionally, we want to slightly penalize each active thruster in order to rank feasible assignments that have the same fuel consumption otherwise.

New Problem Formulation Both penalization terms are only active if some logical conditions are met. We have to introduce binary variables for each thruster in order to express those. The binary variables are set for each thruster with index i individually if the following inequalities are satisfied for $i = [1, ..., n]$

$$\begin{aligned} z_i &= u_{dead} > u_i \\ y_i &= u_i > 0 \\ w_i &= z_i \,\&\, y_i \end{aligned} \tag{2.14}$$

We can stack these integer variables together with the assignment vector to receive a new solution vector $\mathbf{x} \in \mathbb{R}^{4n}$

$$\mathbf{x} = [\mathbf{u}, \mathbf{z}, \mathbf{y}, \mathbf{w}] \tag{2.15}$$

with $\mathbf{u}$ still being the thruster assignment vector containing continuous values. The vectors $\mathbf{z}$, $\mathbf{y}$ and $\mathbf{w}$ are binary valued and have to comply for each thruster with equation 2.14.

Accordingly, the expanded weighting vector $\mathbb{R}^{4n}$ can be formed and is given by

$$\mathbf{f} = [\mathbf{k}_1, \mathbf{0}, \mathbf{k}_2, \mathbf{1} \cdot n \cdot u_{dead}] \tag{2.16}$$

and also the lower and upper bound (given by $\mathbf{lb}$ and $\mathbf{ub}$ respectively) are stacked vectors

$$\begin{aligned} \mathbf{lb} &= [\mathbf{0}, \mathbf{0}, \mathbf{0}, \mathbf{0}] \\ \mathbf{ub} &= [u_{max} \cdot \mathbf{1}, \mathbf{1}, \mathbf{1}, \mathbf{1}] \end{aligned} \tag{2.17}$$

The vectors $\mathbf{0}$ and $\mathbf{1}$ represent constant vectors $\in \mathbb{R}^n$ with only zero or one as entry.

Now a general Mixed Integer Linear Programming (MILP) problem can be formulated

$$\begin{aligned} \min_{x} \quad & \mathbf{f}^T \mathbf{x} && \text{(2.18a)} \\ \text{subject to} \quad & \boldsymbol{x}(\boldsymbol{i}_z, \boldsymbol{i}_y, \boldsymbol{i}_w) \text{ are integers} && \text{(2.18b)} \\ & \mathbf{A}\,\mathbf{x} \leq \mathbf{b}, && \text{(2.18c)} \\ & \mathbf{lb} \leq \mathbf{x} \leq \mathbf{ub}, && \text{(2.18d)} \end{aligned}$$

It becomes apparent that the Vanilla LP formulation is incorporated in the new MILP one. We can see that the fuel consumption is still penalized by the entries in $\mathbf{k}_1$. The thruster assignments $\boldsymbol{u}$ are also upper bounded by u_{max}.

Though, two new penalty terms had to be introduced. The number of active thrusters is penalized for each thruster by $\mathbf{k}_2$. Similar to $\mathbf{k}_1$, it should only contain the same value for all entries. Depending on the ratio of both weights, either the fuel consumption or the number of active thrusters is penalized more heavily. Typically, $\mathbf{k}_2$ should be significantly lower. Secondly, an assignment is heavily penalized as soon as at least one entry in $\mathbf{w}$ is true. We recall that w_i is only unequal to zero if u_i is inside the dead-zone. This penalty corresponds to all thrusters firing with u_{dead}, which means that the assignment will never be considered as the optimal option. Other assignments which have a higher nominal fuel consumption for $\boldsymbol{k}_1^T\boldsymbol{u}$ but in contrast have no assignment $u_i < u_{dead}$ will be preferred.

The constraints are also significantly more complex. Constraint 2.18b defines which entries in the stacked vector $\boldsymbol{x}$ are only allowed to be integers. In our case we even want to have merely binary values for the indexes of $\mathbf{z}$, $\mathbf{y}$ and $\mathbf{w}$. These are represented by $\boldsymbol{i}_z$, $\boldsymbol{i}_y$ and $\boldsymbol{i}_w$. The remaining functionality is implemented by eight different sub-constraints, which are stacked row-wise to form the constraint matrix $\mathbf{A}$ and the corresponding vector $\mathbf{b}$. We denote the different subparts by $\mathbf{A}_1, ..., \mathbf{A}_8$ and $\mathbf{b}_1, ..., \mathbf{b}_8$ respectively and explain their purposes in the next sections.

Constraints for Control Wrench Theoretically, still one equality constraint would be sufficient to enforce an assignment that satisfies equation 2.3. Though, it might be useful if an accepted wrench error can be added directly in the formulation. Thus, we allow a deterministic deviation from the control wrench, which can be used to handle some dynamic problems, as we will see in more detail in later chapters.

The assignment is allowed to actuate a wrench within an artificial lower and upper bound

$$\begin{aligned} \mathbf{B} \cdot \mathbf{u} &\leq \mathbf{T}^G + \mathbf{p}_{upper} \\ \mathbf{B} \cdot \mathbf{u} &\geq \mathbf{T}^G - \mathbf{p}_{lower} \end{aligned} \tag{2.19}$$

Depending on the design goal $\mathbf{p}_{upper}$ and $\mathbf{p}_{lower}$ can be defined freely. A percentage deviation or an absolute value in each wrench dimension could be set in order to explore more fuel conservative wrenches. Paradoxically, even partially larger wrenches might lead to more fuel conservative solutions. It is also possible to allow the thrusters to underactuate a wrench in certain directions, which means the application of lower absolute values. Generally, it has to be taken care of the sign of the different control wrench entries, as upper and lower bound value have then to be switched in the specific dimension for this case. For $\mathbf{p}_{lower} = \mathbf{p}_{upper} = \mathbf{0}$ we get the equality constraint from the LP, as equation 2.19 can only be fulfilled if $\mathbf{B} \cdot \mathbf{u} = \mathbf{T}^G$.

Applied for the expanded solution vector and the MILP formulation we get

$$\begin{aligned} \mathbf{A}_1 &= [\mathbf{B}, \mathbf{0}, \mathbf{0}, \mathbf{0}] \\ \mathbf{A}_2 &= -1 \cdot \mathbf{A}_1 \\ \mathbf{b}_1 &= \mathbf{T}^G + \mathbf{p}_{upper} \\ \mathbf{b}_2 &= -\mathbf{T}^G + \mathbf{p}_{lower} \end{aligned} \tag{2.20}$$

with $\mathbf{A}_1$ and $\mathbf{A}_2$ being $\in \mathbb{R}^{6x4n}$.

Flag Assignments below Dead-Zone We recall that the assignments of every thruster should be compared against u_{dead} and flagged accordingly in the Boolean variable vector $\mathbf{z}$ (see equation 2.14). We have already specified that only binary values are allowed, but now we also have to put up some constraints, which force the solver to set each z_i depending on the value of u_i.

How to generally formulate logical conditions for optimization methods is explained in detail, for example, in [26] or [27]. In [28], methods on how to approach the modeling of certain problems are described and also an overview of the usable algorithms that are compatible with MILP solvers is given. A compact overview of the transformation of several different Boolean operations including comparisons can be found in [29].

To put it briefly, we will use so called Big-M constraints to generate two inequalities, which uniquely set the Boolean variables as desired. We want to express

$$z_i = u_{dead} > u_i \tag{2.21}$$

for each thruster i. If one assumes u_{dead} and u_i to be integer values, we can derive two inequalities with the help of a very big number M

$$0 \leq u_i - u_{dead} + M \cdot z_i \leq M - 1 \tag{2.22}$$

For now we do not put any other condition on the value of M. M only has to be a big number that is always greater than the values in u_i or u_{dead}.

The Big M formulation has to be slightly adopted for the thruster assignment usage, as u_i and u_{dead} contain continuous values. Therefore, the right hand side of the inequality cannot be used in the current state. If an assignment u_i is smaller but very close to u_{dead}, then the constraint pair becomes infeasible. In such cases, the left inequality can only be satisfied if z_i is set to one. However, this leads to a violation of the right hand side inequality. This is not desired and also unintuitive, as it should always be possible to satisfy a comparison flag. We cannot leave out the subtraction part though, as otherwise the strictly greater sign in equation 2.21 is violated. This gets apparent if one sets $u_i = u_{dead}$. Both $z_i = 1$ and $z_i = 0$ would satisfy the given constraint pair.

Therefore, the subtraction by minus one has to be replaced by a customized small constant c_{num}. We can see that only assignments with $u_{dead} > u_i$ are potentially affected by the infeasibility problem. These are the solutions which we try to penalize so that it is already sufficient if $c_{num} < u_{dead}$. Though, generally it is advised to match c_{num} at least with the discretization resolution of the selected signal modulation method. The number can be as close to zero as possible. The lower bound c_{sol} is only given by the solver that has to be able to differentiate it from zero. Thus, c_{num} has to be selected according to

$$c_{sol} < c_{num} < u_{dead} \tag{2.23}$$

Now, we also have to specify the "very big number" M. We know that both u_i and u_{dead} have to be values from the set $[0, ..., u_{max}]$. According to the literature (see e.g. [30]) it is advisable to limit M as much as possible to improve the convergence of the solver. By looking at the inequality pair and by calculating the boundary cases, we can see that z_i is always set correctly if we set M to satisfy

$$M \geq u_{max} + c_{num} \tag{2.24}$$

Thus, we can formulate the two inequality constraints, which are needed in order to enforce equation 2.21 by

$$0 \leq u_i - u_{dead} + (u_{max} + c_{num}) \cdot z_i \leq u_{max} \tag{2.25}$$

Note that we set M to the smallest possible value.

With respect to our stacked solution vector, we can formulate the two constraint matrices and vectors

$$\begin{aligned} \mathbf{A}_3 &= -1 \cdot \mathbf{A}_4 \\ \mathbf{A}_4 &= [\mathbf{I}, (u_{max} + c_{num}) \cdot \mathbf{I}, \mathbf{0}, \mathbf{0}] \\ \mathbf{b}_3 &= -1 \cdot u_{dead} \cdot \mathbf{1} \\ \mathbf{b}_4 &= (u_{dead} + u_{max}) \cdot \mathbf{1} \end{aligned} \tag{2.26}$$

with $\mathbf{A}_3$ and $\mathbf{A}_4$ being $\in \mathbb{R}^{nx4n}$. Accordingly $\mathbf{b}_3$ and $\mathbf{b}_4$ are $\in \mathbb{R}^n$. $\mathbf{I}$ is the $\mathbb{R}^{nxn}$ unity matrix.

Flag Assignments above Zero In order to penalize the number of active thrusters we can take a weighted sum of the active thrusters. Actuated thrusters have an assignment bigger than zero and should be flagged accordingly in the binary variable $\mathbf{y}$. Therefore, we have to set y_i for every thruster to 1 if the corresponding assignment u_i fulfills

$$y_i = u_i > 0 \tag{2.27}$$

Similarly to $\mathbf{z}$ a pair of inequalities can be formulated using big M constraints. Again, we have to exchange the subtrahend in the right hand side M inequality by a small number. We can reuse for example c_{num}. Note however that now, depending on the selection of c_{num}, very small valued assignments for u_i lead to an unsatisfiable constraint.

With an analogous argumentation compared to before, we can also set $M = u_{max} + c_{num}$. This leaves us with the inequality pair

$$0 \leq -u_i + (u_{max} + c_{num}) \cdot y_i \leq u_{max} \tag{2.28}$$

Once again, we can give the inequality pair in standard matrix and vector formulation with respect to the stacked vector $\mathbf{x}$

$$\begin{aligned} \mathbf{A}_5 &= -1 \cdot \mathbf{A}_6 \\ \mathbf{A}_6 &= [-\mathbf{I}, \mathbf{0}, (u_{max} + c_{num}) \cdot \mathbf{I}, \mathbf{0}] \\ \mathbf{b}_5 &= -1 \cdot \mathbf{0} \\ \mathbf{b}_6 &= u_{max} \cdot \mathbf{1} \end{aligned} \tag{2.29}$$

We have the same dimensions of the matrices and vectors as before with $\mathbf{A}_5$ and $\mathbf{A}_6$ being $\in \mathbb{R}^{nx4n}$. Note the change in positioning of the non-zero sub-matrices. Also $\mathbf{b}_5$ and $\mathbf{b}_6$ are $\in \mathbb{R}^n$. $\mathbf{I}$ is still a unity matrix $\in \mathbb{R}^{nxn}$.

Flag Assignments inside Dead-Zone Area It is not desired to penalize assignments which contain zero values. Actually, we want to fire as little thrusters as possible. Thus, it is not sufficient to use the Boolean entries in $\mathbf{z}$ to selectively penalize assignments in order to embed the minimal valve opening time. We have to make sure that the zero assignment is specifically not included.

This is why the additional vector variable $\mathbf{y}$ was introduced. Together with $\mathbf{z}$ we can generate

$$w_i = z_i \,\&\, y_i \tag{2.30}$$

which expresses a binary AND. For each thruster w_i is only true, if both of the previously introduced binary entries in $\mathbf{z}$ and $\mathbf{y}$ are true in the corresponding entry.

A Boolean AND can be simply expressed by

$$0 \leq z_i + y_i - 2 \cdot w_i \leq 1 \tag{2.31}$$

in integer linear programming, which means that we get another additional constraint pair. We already have the fixed and the variable parameters separated so that we can immediately write the two constraints in the stacked matrix and vector form

$$\begin{aligned} \mathbf{A}_7 &= -1 \cdot \mathbf{A}_8 \\ \mathbf{A}_8 &= [\mathbf{0}, \mathbf{I}, \mathbf{I}, -2 \cdot \mathbf{I}] \\ \mathbf{b}_7 &= -1 \cdot \mathbf{0} \\ \mathbf{b}_8 &= \mathbf{1} \end{aligned} \tag{2.32}$$

The dimensions of both the matrices and the vectors are still $\mathbb{R}^{nx4n}$ and $\mathbb{R}^n$ respectively.

Evaluation We finally obtained a formulation which includes all the important thruster properties. It was shown that this can be done by converting logical conditions into integer linear programming formulations. With this method the most fuel conservative assignment, which also ideally actuates the demanded control wrench, will be received. This assumes that the MILP solver converges to this global optimum. A meaningful allowed wrench deviation was also added by changing the equality to two inequality constraints.

The price one has to pay for this lies within the necessity of converting the LP to a MILP problem, which requires a different class of solvers. The number of parameters is quadrupled and a number of new constraints have to be added. The computational load will be significantly higher, as we will see in Section 2.5.

As a small note, it can be added that the run-time can be optimized if one also adds knowledge from the selected signal modulation method to the MILP formulation. If we know, for example, that a Pulse-Width Modulation (PWM) is used, we can use the limited discretization of the modulator due to limitations in possible PWM frequencies, resulting in u_{disc}. More details about the PWM and its application in the thruster actuation can be found in Subsection 2.4.2, but we can optimize the run-time by adapting $\mathbf{b}_5$ from equation 2.29. Now, we do not compare against zero, but change it to a value smaller than u_{disc}. By selection of u_{disc} as a boundary, we get

$$\mathbf{b}_5 = -1 \cdot -u_{disc} \cdot \mathbf{1} \tag{2.33}$$

With this adaption we basically allow assignments in the range $0 < u_i < u_{disc}$ again, as they these assignments will not be penalized by $\mathbf{w}$. Any assignment within this range after the selection process will obviously be set to zero in the preparation step before the signal modulator, but the enlargement of feasible assignments above zero greatly reduces the convergence time. Also, as we will see in a later chapter, the control system will be more robust against uncertainties in the model. Overall, we make only a minimal additional wrench error.

2.3.4 Vanilla Linear Programming Formulation With Counter-Thrust Notion

The main concern of the MILP formulation lies within the increased complexity and the resulting increase in computational cost. A slower convergence to the optimal solution is expected due to the more complex solution space. In the following formulation we follow the goal of adapting the regular LP problem statement in such a way that it then adheres to the specification of the minimal valve opening time. In thrust space this is realized if no assignments between $0 < u_i < u_{dead}$ are present. Thus, we combine the advantages of both the standard LP and the MILP.

We now assume that a special class of *antagonistic* thruster configurations is used. Note however that the idea of the following notion could be also adapted to general kinds of configurations.

We assume that every thruster in the configuration has exactly one antagonistic thruster which annihilates its applied wrench. We can identify corresponding pairs in a given mapping matrix $\mathbf{B} \in \mathbb{R}^{6xn}$ and flag them in a sparse matrix $\mathbf{B}_{count} \in \mathbb{R}^{nxn}$. Each row and column is associated with the corresponding thruster of the same index.

We build this matrix by starting with an identity matrix. Afterwards, we go through all thrusters and find its antagonist. Let us imagine that thruster one and thruster four cancel each other out. This means, we will not get an actuated wrench if $u_1 = u_4$ and $u_i = 0$ for $i \in [2, 3, 5, ..., n]$. Thus, we have to set the entries (1,4) and also (4,1) in $\mathbf{B}_{count}$ to 1. Similarly, this is done for all other thrusters. The rows and the columns of $\mathbf{B}_{count}$ have to sum up to two in the end.

$$B_{count}(i,j) = \begin{cases} 1, & \text{for } i = j \\ 1, & \text{thruster j is antagonist of i} \\ 0, & \text{otherwise} \end{cases} \tag{2.34}$$

$\mathbf{B}_{count}$ is used in the following to selectively transform an infeasible thruster assignment $\mathbf{u}_{old}$, which ignores t_{mib}, to a feasible assignment $\mathbf{u}_{new}$ by addition of a simple null space thrust assignment. In order to do this, we have to first introduce the assignment vector $\mathbf{u}_{add}$. The entries can be calculated by checking whether any of the preliminary assignments $u_{i,old}$, after conducting the LP, are within the bound $0 < u_{i,old} < u_{dead}$. Depending on the result of this comparison, we fill a vector

$$u_{i,add} = \begin{cases} u_{dead}, & 0 < u_{i,old} < u_{dead} \\ 0, & \text{otherwise} \end{cases} \tag{2.35}$$

for all thrusters $i \in [1, ..., n]$.

The transformation itself is then achieved with

$$\mathbf{u}_{new} = \mathbf{u}_{old} + \mathbf{B}_{count} \cdot \mathbf{u}_{add} \tag{2.36}$$

This operation allows us to generate relative thrusts, which are smaller than u_{dead}. In the previous example a small assignment for u_1, which is inside the dead zone, will lead to a minimal actuation of u_4 and a proper actuation of u_1. Thus, the relative thrust is only shifted to a feasible solution. It is obvious that the optimal solution from the LP formulation will always assign only one of the two antagonistic thrusters a value unequal to zero.

A small adaption of equation 2.35 is also reasonable to reduce fuel usage in the real system

$$u_{i,add} = \begin{cases} u_{dead}, & u_{disc} < u_{i,old} < u_{dead} \\ 0, & \text{otherwise} \end{cases} \tag{2.37}$$

Here, the limited discretization of e.g the PWM is respected. The variable u_{disc} specifies the smallest thrust value that is not mapped to zero due to the discretization.

In general, the transformed assignment $\mathbf{u}_{new}$ is not guaranteed to be the most fuel conservative of all feasible ones and thus not always equal to the MILP result in this regard.

When thinking about the possibility of introducing wrench errors due to this operation, we only have to look at the boundary u_{max}. We will not distort the actuated wrench as long as the following condition is met

$$u_{dead} \leq \frac{u_{max}}{2} \tag{2.38}$$

Generally, it is advised to use this adaption only together with a signal modulation method that has synchronized pulse intervals across all thrusters (e.g. a PWM). Namely, a usage together with a Pulse-Width Pulse-Frequency (PWPF) modulator is not meaningful. A list of selected modulators is presented in more detail in Section 2.4.

2.3.5 Quadratic Programming Formulation

All the previously presented optimization methods have the problem that they will break down in case of control wrenches located outside of the controllability envelope. In such cases, the constraints stemming from equation 2.3 can never be satisfiable.

This raises the question of how to proceed in such cases. The easiest possibility would be to just not actuate anything if we cannot realize the control wrench with a corresponding thruster assignment. The most intuitive approach would be to actuate the wrench, which is the closest possible one. We want to minimize the distance, typically in the form of optimizing the quadratic reconstruction error

$$\begin{aligned} \min_{\mathbf{u}} \quad & \frac{1}{2}(\mathbf{Bu} - \mathbf{T}^G)^T \mathbf{W} (\mathbf{Bu} - \mathbf{T}^G) \\ \text{subject to} \quad & 0 \leq u_i \leq u_{max} \quad \text{for } i \in [1...n]. \end{aligned} \tag{2.39}$$

with the weighting matrix $\mathbf{W} \in \mathbb{R}^{6x6}$. Typically, it will contain only diagonal entries, which are required in order to relate torque and force errors accordingly.

The resulting assignment of this optimization could be directly used, but it is not optimized with respect to the fuel consumption. Especially with respect to wrenches inside the controllability envelope, there might exist multiple thruster assignments which realize the minimal distance wrench. We have to add the linear fuel optimization term in the Quadratic Programming (QP) formulation to jointly optimize fuel and wrench distance. The result of the joint optimization is given by

$$\begin{aligned} \min_{\mathbf{u}} \quad & \frac{1}{2}(\mathbf{Bu} - \mathbf{T}^G)^T \mathbf{W} (\mathbf{Bu} - \mathbf{T}^G) + \mathbf{k}^T \cdot \mathbf{u} \\ \text{subject to} \quad & 0 \leq u_i \leq u_{max} \quad \text{for } i \in [1...n]. \end{aligned} \tag{2.40}$$

We are now jointly optimizing both the distance and the fuel consumption. Thus the weight matrix $\mathbf{W}$ and the weight vector $\mathbf{k}$ have to be adjusted accordingly. Dependent on the relative values of both weights, a more fuel conservative wrench with a higher deviation to the control wrench is actuated.

By expansion, one can obtain a standard QP formulation

$$\begin{aligned} \min_{\mathbf{u}} \quad & \frac{1}{2}\mathbf{u}^T\mathbf{B}^T\mathbf{W}\mathbf{B}\mathbf{u} + \left(\mathbf{k} - \mathbf{T}^G\mathbf{W}\mathbf{B}\right)^T\mathbf{u} \\ \text{subject to} \quad & 0 \leq u_i \leq u_{max} \quad \text{for } i \in [1...n], \\ & |\mathbf{B}\,\mathbf{u}| \leq |\mathbf{T}^G, \\ & sign(\mathbf{B}\,\mathbf{u}) = sign(\mathbf{T}^G), \end{aligned} \tag{2.41}$$

Note that an additional constraint was added. Depending on the configuration, there is no guarantee that the some dimensions are not overactuated or that the sign remains unchanged. For example, if one imagines a very large force in x-direction, the minimal distance inside the controllability envelope might only be obtained by actuating forces and wrenches in the other directions. This is an undesired behavior for a lot of controls (e.g. attitude control). The two constraints could also be better expressed by two inequality constraints with an implicit handling of the signs. Shortly we can identify two cases for both constraints.

$$\mathbf{B}_i \cdot \mathbf{u} \begin{cases} \leq T_i \\ \geq 0 \end{cases} \quad \text{for } T_i \geq 0 \qquad \begin{cases} \leq 0 \\ \geq T_i \end{cases} \quad \text{for } T_i < 0 \tag{2.42}$$

2.3.6 Pre-Computed Tables

In the past, a common approach to solve the thruster assignment problem was to store a set of characteristic thruster assignments in tables. The main advantage here lies within the simplicity of the onboard computation, as only the best suited assignment has to be selected from the set of pre-computed ones. The onboard run-time also is deterministic compared to optimization methods which have varying run-times. Often, thrusters are grouped together so that they are regarded as one actuator in order to achieve, for example, pure torques. Alternatively, also a set of maneuvers can be stored and selected on a case by case basis depending on the input command.

The general pre-computed table approach is not applicable for more complex spacecraft geometries, which have coupling effects. This means that it heavily constraints the thruster configuration design to only axis-symmetric layouts. Martel [31] proposed an adaption to the catalogue table design for the 6 degrees of freedom actuation, which bypasses this limitation. Additionally, the resulting table adheres to given constraints like a limited amount of active thrusters at the same time. In [32] this solution was adopted to the 3 DoF attitude control problem.

From a naive point of view, we could just store all possible assignments in a hash table. For example, in a PWM modulated thruster only a limited number of discretizations of the continuous assignment can be done. In theory, one could simply use a distance measure, like the weighted squared distance of the wrench, for all possible assignments. If multiple assignments have the same distance, we can rank them with respect to the previously stated performance parameters like fuel consumption and active thrusters. However, the number of all possibilities is incredibly high

$$n_{disc}^{n} \tag{2.43}$$

with n_{disc} being the number of discretization steps in n PWM actuated thrusters. If we only consider assignments which have at most 6 thrusters active, we still have

$$\frac{n!}{6! \cdot (n-6)!} \cdot n_{disc}^{6} \tag{2.44}$$

combinations. The amount of possibilities only become feasible when we in addition restrict ourselves to just consider an on and off discretization

$$\frac{n!}{6! \cdot (n-6)!} \cdot 2^{6} \tag{2.45}$$

Alternatively, one could consider a discretization of the wrench space. We first calculate the most optimal assignments offline according to a MILP formulation, which considers all important thruster properties, but only for a limited number of wrenches inside the wrench space. Onboard, the operations reduce to the task of finding the closest calculated wrench with some kind of distance measure. In order to reduce the required amount of comparisons, we can use similarity estimations and hashing. Still with this simple approach we have high memory requirements due to the high dimensionality.

2.3.7 Combinations

Approaches with the goal to combine the benefits from both the run-time efficiency of pre-computed tables and the more fuel efficient and wrench-optimal selections of thrusters of optimization methods have been proposed. A two-step algorithm based on a geometrical representation of the problem was developed in [20]. For nominal operation, the best combination of six thrusters is derived from a binary tree structure, which is stored in the onboard memory. On-board only a few scalar products have to be computed in order to reduce the amount of assignment candidates by moving along the binary tree. If the control wrenches are relatively large, a second step is required in order to minimize the wrench error. A special modification of the Simplex method is used in order to iteratively reduce this error.

2.4 Signal Modulation

In this section we will focus in more detail on the most common approaches, which are conducted in order to translate the continuous valued thruster assignments to sequences of on-off pulses. At fixed time intervals, given by the control frequency f_{ctrl}, we have to calculate a new thruster assignment, which is then actuated by the modulation. We denote the desired thruster assignment, coming from the assignment block, with $\mathbf{u}$. The translated time-dependent pulse sequence is depicted with $\tilde{\mathbf{u}}(t)$. In the end, we are more interested in the effectively actuated thrust during the control time interval, which is given by $\overline{\mathbf{u}}$. It can be calculated from the average pulse during the time interval

$$\overline{u}_i(k) = \frac{1}{t_{ctrl}} \int_{(k-1)\cdot t_{ctrl}}^{k\cdot t_{ctrl}} \tilde{u}_i(t)dt \tag{2.46}$$

for the i-th thruster and the k-th control time interval. We will drop the dependency on k in the notation for $\overline{u}_i(k)$ from now on.

Note that there exist various sources also with respect to spacecraft control that focus in more detail on the different types of thruster modulation (e.g. [12] or [33]). As this is not the main topic of this work, we will only shortly introduce some of these main approaches. These will be adapted so that they are coherent with our assumptions with respect to the used thrusters and so that they can be used as a sub-block within our thruster actuation block with wrench-interface.

2.4.1 Bang-Bang Modulation

The most straightforward approach, often called Bang-Bang modulation, is to fully actuate any thruster, which has an assignment above zero. These thrusters then fire for the complete control time interval. In this case we have $\tilde{\mathbf{u}}(t) = \overline{\mathbf{u}}$.

$$\overline{\mathbf{u}} = \begin{cases} u_{max}, & \text{if } u_i > 0 \\ 0, & u_i \leq 0 \end{cases} \tag{2.47}$$

There is only no actuation, if all thrusters have a zero assignment. As this is rarely the case in a controlled system, especially with on-off-thrusters, we will have an ongoing actuation. Implicitly, the minimal valve opening time t_{mib} is respected as long as the control frequency does not violate the inequality

$$f_{ctrl} \leq \frac{1}{t_{mib}} \tag{2.48}$$

In order to reduce the amount of active thrusters and thus also the fuel consumption, we can artificially introduce a cut-off assignment value u_{cut}. Thrusters with assignments below this value will not be actuated.

Note that the literature also often refers to this adaption as dead-zone (see e.g. [12]). However, this work already introduced the concept of dead-zone for u_{dead} as the assignment value, which results in a thruster firing for t_{bmin}. This is included in the Bang-Bang modulation if the inequality in equation 2.48 is fulfilled. Therefore, $u_{dead} = 0$ in the Bang-Bang modulation.

$$\overline{\mathbf{u}} = \begin{cases} u_{max}, & \text{if } u_i > u_{cut} \\ 0, & u_i \leq u_{cut} \end{cases} \tag{2.49}$$

From our limited point of view of the signal modulation being a simple input-output block, we cannot make any assumptions regarding the dynamic stability. However, we can see that this kind of modulation does not distinguish between the absolute value of the different assignments. This will lead, even with the extension of the cut of value, to a large wrench error. Especially for the general case of coupled entries in the applied wrench, a stable control might be hard to find. For attitude control, where the three principal axes are under some assumptions nearly uncoupled, one can show that this kind of modulation will always converge to a low-frequency limit cycle if coupled with a suitable controller ([15], chapter 3).

In the following work, we will not investigate the properties of the closed loop Bang-Bang modulation separately. Though for a special parameter selection we can, as we will see later on, relate the PWM setup with a Bang-Bang control with cut-off.

2.4.2 Pulse-Width Modulation

Pulse-Width Modulation is often used in engineering applications in order to translate a continuous signal into a sequence of switched two-valued pulses. By adjusting the duration of the high-pulse, we can modulate the amplitude of the signal, which is applied on average and which is proportional to the input.

Intersective Technique There exist numerous approaches on how to realize a pulse-width modulated signal. One example is the intersective technique, which uses a saw-tooth signal. The saw-tooth runs with a fixed frequency and is compared against the time-dependent control signal. The binary output is set to a high state, if the sawtooth is below the control signal. Analogously, it is set to low in the opposite case. Figure 2.3 depicts how the output is generated for a sinusoidal input signal.

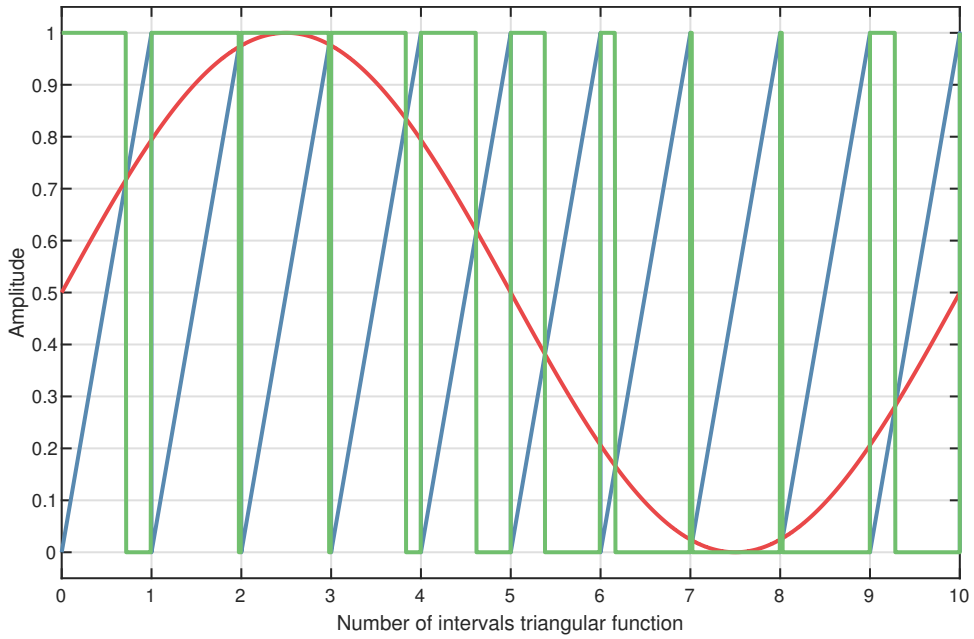


Figure 2.3: PWM for a sinusoidal input (red) with intersective technique using a saw-tooth function (blue) to generate binary pulses (green)

Application in Thruster Environment For an application with thrusters, we have to slightly adjust this technique. We will update the thruster assignments regularly depending on the control time interval t_{ctrl}. During this control time interval, we want to apply a signal, which on average is very close to the assigned, constant value, which is obtained at the start of each interval. In addition, we cannot actuate a pulse shorter than t_{mib}.

In the PWM we manipulate the time domain, instead of proportionally adjusting the amplitude of the signal. Hence, we can calculate the minimal assignment in thrust space, which is required to fire longer than t_{mib}. This parameter was already introduced abstractly in the previous section and named u_{dead}.

$$u_{dead} = \frac{t_{mib}}{t_{ctrl}} \cdot u_{max} = f_{ctrl} \cdot t_{mib} \cdot u_{max} \tag{2.50}$$

Assignments below this parameter are not applicable with a PWM and will be set to zero. As a partial result, we also observe that the control frequency is limited in the PWM as well to

$$f_{ctrl} \leq \frac{1}{t_{mib}} \tag{2.51}$$

Otherwise no thruster can be actuated. A thruster would have to continuously fire across multiple time intervals and this cannot be guaranteed at this point, as each control time interval is independent in our structure. With a PWM, the control frequency is bounded by the thruster switching frequency.

Now, we also want to limit the number of possible pulse durations in order to reduce the computational load resulting from the PWM. We introduce the update rate of pulse durations, denoted by a frequency f_{pwm}. Depending on the fraction of both control and PWM frequency, we can calculate the number of discretization steps.

$$n_{disc} = \frac{f_{pwm}}{f_{ctrl}} \tag{2.52}$$

Obviously, we demand that $f_{pwm} \geq f_{ctrl}$. Additionally it makes sense to synchronize the frequencies, so that n_{disc} is an integer value. We can specify the minimal relative assignment u_{disc} by

$$u_{disc} = \frac{f_{ctrl}}{f_{PWM}} \cdot u_{max} \tag{2.53}$$

Lastly, we also want to center the pulse around the half-time of the control time interval. This will not change the relative firing times but improve the dynamic behavior of the system and was recommended by [34]. A splitting of the pulse into several evenly distributed smaller pulses will not be done, as it brings only a marginal improvement at the cost of more valve wear down.

Our developed PWM block will first check against u_{dead} and will set any u_i to zero if it is below. We then translate the assignment into a number of on pulses with

$$n_{on} = floor\left(n_{disc} \cdot \frac{u_i}{u_{max}}\right) \tag{2.54}$$

Note that we round towards the next lowest integer value here. This means that only $u_i = u_{max}$ will result in a full firing period. Other forms of rounding might be also sensible though.

To realize the firing logic, we now only have to count the current PWM time interval value, which is cyclically reset after each update of the thruster assignments. From the number of on pulses, we can derive a logic table, which specifies for which PWM time interval index the different thrusters have to be switched on. Note that this logic table is shifted and centered around the PWM half-time interval.

Figure 2.4 shows a depiction of the presented PWM switching logic. Strictly speaking, the thruster assignment is not part of the modulation but is only included to stress the coupling of the time intervals of the control and the pulses.

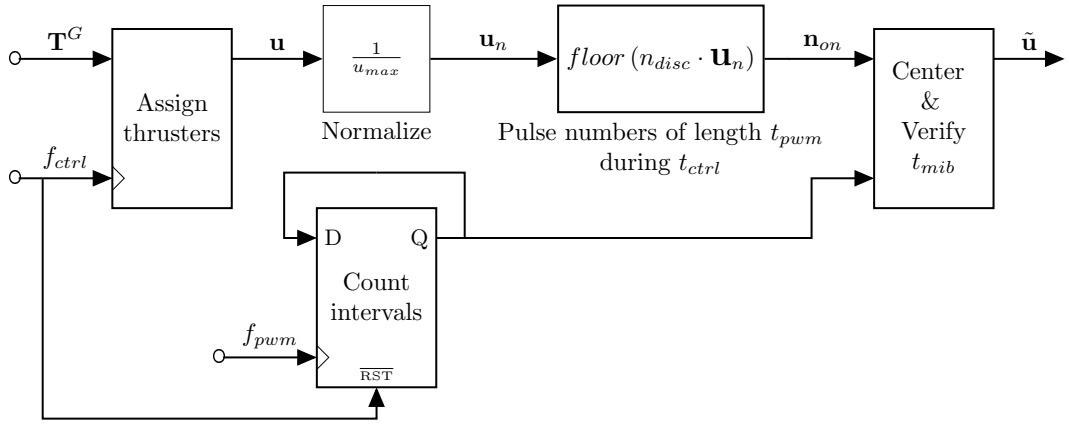


Figure 2.4: Block diagram of the used PWM adaption

Relating PWM and Bang-Bang The Bang-Bang modulation with cut-off can be modeled by a specially parameterized PWM. The PWM signal now only has one discretization step, which means that $f_{pwm} = f_{ctrl}$, resulting in either no or full firing. Now we basically realized a Bang-Bang modulation with $u_{cut} = u_{max}$. For different values of u_{cut}, we have to tune the gains of the controlled system accordingly. Note that this is no exact relation, as due to the redundancy of the configuration a scaled wrench does not necessarily result in a scaled assignment. However for $f_{ctrl} = \frac{1}{t_{mib}}$ we have an exact relation.

Vibration Damping Other modulators are often preferred to the PWM, as the coupled actuation and input time intervals can introduce vibrations to the system. One could also adopt the PWM implementation, so that the pulses of the thrusters are distributed across the control time interval in a way that minimizes induced vibrations. A random positioning of the pulses independent for each thruster in the time domain might prove to be useful.

2.4.3 Pulse-Width Pulse-Frequency Modulation

The PWM uses a constant modulation frequency f_{pwm} in order to translate a continuous valued signal, fixed with respect to time, into a binary signal with proportional pulse lengths.

The Pulse-Width Pulse-Frequency (PWPF) modulator does not only adjust the pulse lengths but also the frequency of the pulses. We equated the update rate of new assignments with the control frequency. This means that we can now have either multiple, one or no firing per control time interval, as the pulse frequency is now not directly related to it. Additionally, also the different thrusters are now not synchronized with each other but have independent firing intervals.

This property is especially useful if the dynamics of the spacecraft contain structures with low damping coefficients. If some eigenfrequencies of the system are close to the control frequency, it is possible that we, by accident, excite the structure [17]. Vibration reduction (see [35] or [36]) and suppression [37] are the reason why PWPF modulators are often selected. For the PWPF, in general very large and very small assignments for a thruster will lead to a low pulse frequency.

The decoupling of the actuation and control time intervals also allows to, in principle, choose the control frequency freely. Now, it is not limited by the thrusters and can be optimized with respect to other parameters, like sensor rate, stability analysis or computational load. In a sense, sensing and control are decoupled from the actuation. The control frequency only has an impact on how frequently the input of the PWPF modulator is updated.

Our limited static point of view partially breaks down for the design of this type of modulator, as we cannot give any statements regarding the static wrench error. We defined this in the context of occurring during one control time interval. Due to the decoupling, only a closed-loop analysis can identify this parameter. In the following we can still derive some valuable properties by using the input-output point of view though.

General Structure As visible in Figure 2.5, the PWPF is realized by a feed-back loop, which consists of both a pre-filter of first order and a Schmidt Trigger. The Schmidt Trigger can be seen as an on-off relay with an additional hysteresis and a dead-band.

Once the input $f(t)$ is higher than d, the output is set to U_m. Otherwise, if $f(t)$ is below $d - h$, the output goes back to zero. $U_m(t)$ is fed back and subtracted from the systems input $r(t)$. The result $e(t)$ is the input of the Pre-Filter. Obviously, the modulator is non-linear in total but can be assumed linear with a correct parameter selection as we will se later.

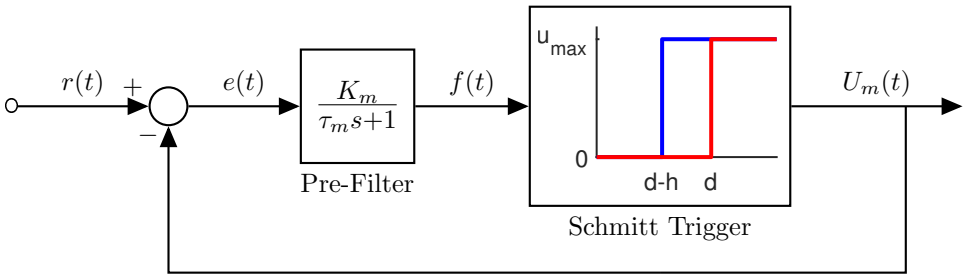


Figure 2.5: Block diagram of a general PWPF modulator (adapted from [38])

The PWPF modulator has four open parameters, compared to the PWM, for which only f_{pwm} has to be chosen. They are shortly listed in the following with a small description of their effect.

- d: Switching condition from 0 to U_m of the Schmidt Trigger

- h: Switching condition from U_m to 0 of the Schmidt Trigger
- K_m: Gain of the Pre-Filter
- τ_m: Time constant of the Pre-Filter

These parameters have to be selected in a meaningful way, as they change the closed-loop behavior of the modulator. This makes it more dependent on the selected control and as a result the PWPF modulator is more difficult to design in contrast to the PWM. Some implementations have an additional pre-gain before the input of the PWPF.

In the literature, the parameters of the PWPF modulator are often treated as design parameters for the complete closed-loop system. It got attention especially in the context of attitude control and there exist some recommendations with respect to optimal PWPF parameters. All, [35], [37] and [39] conducted static and dynamic analysis and derived some recommended values, which are slightly different from each other but generally in very similar ranges. In [40] a joint tuning of both control and PWPF parameters with the help of particle swarm optimization is done.

At this point however, we do not want to limit ourselves to a specific type of control. We will derive our own set of recommendations for the design of the PWPF parameters with respect to the given thruster properties.

Static Relations First, we will conduct a static analysis. We make the assumption that the we have a constant input value $r(t) = r_c$. With this premise, the modulator becomes system independent. Some properties of the static case are also valuable for the dynamic behavior if the dynamic change of $r(t)$ is slow compared to the pulse time intervals. The given relations closely follow the ones given in [38] in both notation and structural approach.

The pre-filter is a regular filter of first order. Its behavior in time is analogously to an electric capacitor connected with a resistor in series, which is charged by a current through a constant voltage. As long as the error $e(t) = r(t) - U_m(t)$ stays constant, we can drop the time dependency and have a behavior specified by the characteristic time constant

$$f(t) = f(0) + (K_m e - f(0))(1 - e^{-\frac{t}{\tau_m}}) \tag{2.55}$$

with $f(t \to \infty) = K_m e$.

In order to generate a sequence of on-off pulses, the switching conditions have to be met. This means that we have a minimal and maximal input value, r_{min} and r_{max} respectively. Only within this operation area a quasi-linear operation is observed. If we assume that the output is in off-state $U_m(0) = 0$ at the beginning of the operation, we can see that the minimal input to activate the Schmidt-Trigger for $t \to \infty$ can be calculated by

$$r_{min} = \frac{d}{K_m} \tag{2.56}$$

Analogously, we can derive the switch off condition for $U_m(0) = U_m$

$$r_{max} = U_m + \frac{d-h}{K_m} \tag{2.57}$$

Outside of this operation area either a continuous high pulse for $r > r_{max}$ or a zero pulse for $r < r_{min}$ will be given.

For a constant input, Figure 2.6 visualizes how the different signals within the PWPF modulator are changed with respect to time.

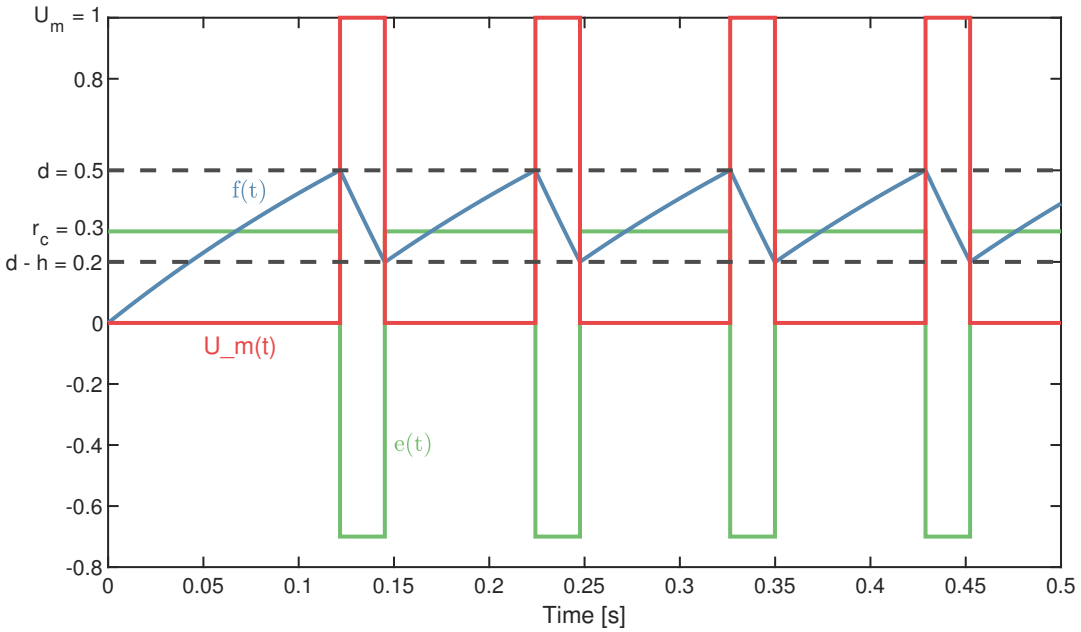


Figure 2.6: Switching times of the PWPF modulator for constant input $r_c = 0.3$ (with $d = 0.5$, $h = 0.3$, $\tau_m = 0.3$, $K_m = 5$)

The on-time of the modulator is characterized by the time that is needed in order to change the relay input from $f(0) = d$ to $f(t_{on}) = d - h$. Given this knowledge, we can plug into equation 2.55 and solve for t_{on}

$$t_{on} = -\tau_m \ln\left[1 + \frac{h}{K_m\left(r_c - U_m\right) - d}\right] \tag{2.58}$$

In a similar fashion, the off-time of the modulator can be calculated. In this case, the transition time from $f(0) = d - h$ to $f(t_{off}) = d$ is of specific interest

$$t_{off} = -\tau_m \ln\left[1 - \frac{h}{K_m r_c - (d-h)}\right] \tag{2.59}$$

It has to be stressed that for $d \neq h$, this equation holds only for the tuned in state. The very first operation requires an additional off-time, which is needed to integrate the relay input from zero to $d - h$.

With both timings, we can specify the pulse frequency f_{PWPF} and also the so-called Modulation Factor (MF), which are defined by

$$f_{PWPF} = \frac{1}{t_{on} + t_{off}} \tag{2.60}$$

and

$$MF = \frac{t_{on}}{t_{on} + t_{off}} \tag{2.61}$$

By plugging in the specific relations for t_{off} and t_{on} into both definitions, we can see the previously mentioned properties. The pulse frequency will be not only dependent on the input but also changes with respect to the other open parameters. It becomes apparent why we talked about a pseudo-linear behavior of the PWPF, as we see that the modulation factor is in fact not a linear function.

Adaption to Thruster Application We recall that in its current form we have four open parameters in the PWPF modulator, specifically K_m, τ_m, d and h. By using the constraints given by the thrusters, we can reduce the number of open parameters. This makes the tuning with respect to the closed-loop system easier.

By investigating equation 2.58 and using the property of the natural logarithm being a non-decreasing function, we can see that the minimal on-time is obtained for $r = r_{min}$. In order to adhere to the minimum impulse bit t_{mib}, we demand that

$$t_{mib} = -\tau_m \ln\left[1 + \frac{h}{K_m\left(\frac{d}{K_m} - U_m\right) - d}\right] = -\tau_m \ln\left[1 - \frac{h}{K_m \cdot U_m}\right] \tag{2.62}$$

This equation can be solved with respect to τ_m. This parameter is now fixed, as it directly unfolds by selection of the other three parameters.

In the next step, we have a look at equations 2.57 and 2.59 and will fix another open parameter in a way, which leaves us with only one tuning parameter for the PWPF modulator. Note that this parameter selection, however, limits the possibilities of adjusting the PWPF modulator to some degree.

We could in theory also stop at this point and either optimize K_m, d and h jointly with the control parameter or use recommended values from the literature.

Ideally, we want to have a zero off-time for $r_c = U_m$. However, we can see that here the effect of the non-linearity has an impact. The shortest off-time is equal to t_{mib} and is obtained for r_{max}. In order to minimize the off-time for maximal input values, we have to set

$$d = h \tag{2.63}$$

This approach uses the fact that for our application $r_c \leq U_m$.

Due to the fixation described in equation 2.63, we can simplify the two characteristic equations of the PWPF modulator, namely the on- and off-time, by

$$\begin{aligned} t_{on} &= \frac{t_{mib}}{\ln\left[1 - \frac{r_{min}}{U_m}\right]} \cdot \ln\left[1 + \frac{r_{min}}{r_c - U_m - r_{min}}\right] \\ t_{off} &= \frac{t_{mib}}{\ln\left[1 - \frac{r_{min}}{U_m}\right]} \cdot \ln\left[1 - \frac{r_{min}}{r_c}\right] \end{aligned} \tag{2.64}$$

with only having a dependency on r_{min} and the input r_c. Note that also equation 2.62 was used in order to replace τ_m. Selecting r_{min} as open parameter is very intuitive when coming from the PWM, as it is analog to the notion of dead-zone u_{dead}.

Assuming a correct pre-procession of our assignments, we know that the inputs of the PWPF are bounded to $0 \leq r_i \leq u_{max}$ in our application with thrusters. Without loss of generality for the PWPF modulator, we can also scale the thruster assignments by the factor $\frac{1}{u_{max}}$ before feeding them into the modulator so that $0 \leq r_i \leq 1$. Accordingly the output is binary with $U_m \in [0, 1]$, which indicates if a thruster is switched on or off. Obviously, the applied thrust on the output has to be scaled up again by the factor u_{max}, when calculating the applied wrench for an update of the internal states.

In the following, we want to shortly analyze how different selections of r_{min} affect the behavior of the static PWPF modulator for different inputs r_c. This is depicted in Figure 2.7. Note that only inputs which are located within the pseudo-linear range $r_{min} < r_c < r_{max}$ of the modulator are visualized. As soon as either the on- or off-timing is infinite or not defined according to equations 2.64, no value is plotted.

In general, we can see that values close to r_{min} result in short on-timings and the highest off-time. The pulse frequency and the MF is symmetric to $\frac{r_{max}+r_{min}}{2}$. The maximal frequency is obtained at this point and decreases for higher distances. The modulation factor rises with larger values for r_c.

We can also observe that the non-linear behavior of the PWPF modulator is more pronounced for larger r_{min}. The modulation factor is not linear anymore but heavily skewed for r_{min} close to 1. With an infinitely small selection of r_{min} we would obtain a perfect linear behavior between in- and output. However, the pulse frequencies also go down with respect to a lower r_{min}, which can be seen by the shorter on- and off-times for this parametrizations as well. This heavily affects the dynamical behavior of the modulator, as we will see later.

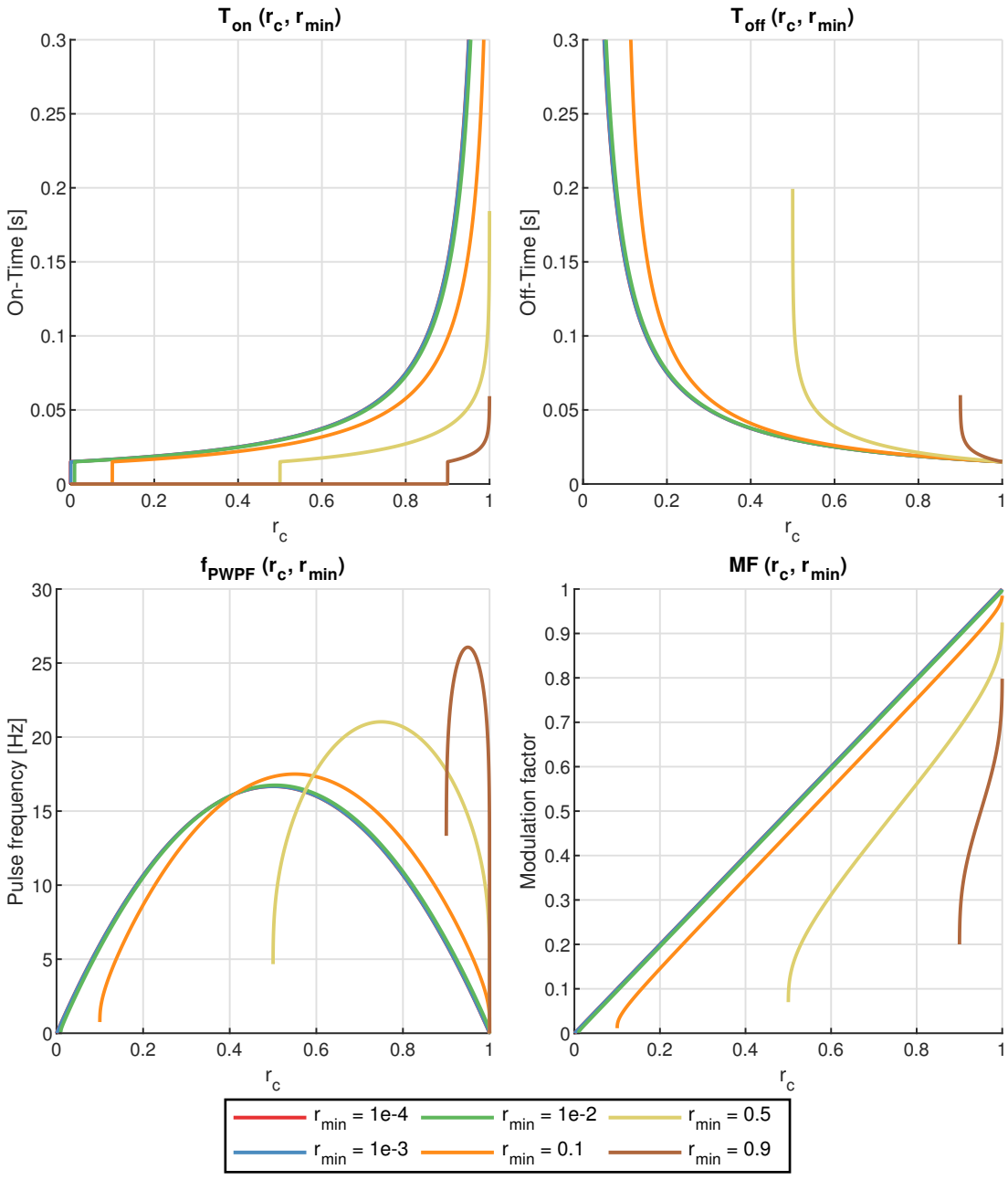


Figure 2.7: Characteristic relations of the PWPF modulator with respect to a variation in r_{min}

As a final remark, it should be noted that the limitation in the parametrization variety of the PWPF modulator, due to selecting $d = h$, does not heavily limit the behavior of the PWPF modulator. The given plots in Figure 2.7 reproduce the general output of a more complex PWPF modulator with K_m, d and h as open parameters.

Dynamic PWPF Modulator Until now it was assumed that the input $r(t)$ is constant. However, in the closed loop system the assignments will change periodically depending on f_{ctrl}.

If some thrusters have very low assigned thrusts at the start of one control time interval, even for assignments above r_{min} it is not guaranteed that these fire during the control time interval. This might lead to a completely different assignment in the next time interval, as it is not constant but changes in practice according to a stair-case function. We can see that this parameter has to be optimized within the closed-loop and represents a trade-off between linear modulator behavior and firing frequencies of the PWPF.

As previously mentioned, the control timings are in no way synchronized with the different pulse frequencies of the thrusters. This means that even though the minimal valve opening time is considered and followed for constant inputs $r(t) = r_c$, a changing $r(t)$ might lead to a violation of this constraint.

Let us assume that an initial input is equal to $r(0) = r_{min}$ or also slightly above this threshold. Directly after the firing is initiated by the PWPF, the input value is updated to $r(t_1) = 0$. This will lead to a faster unloading of the input of the Schmidt-Trigger by the pre-filter so that the shut-off condition is reached quicker.

Such a behavior can in theory be observed for all possible parameter selections, assuming we set up the PWPF modulator in the previously developed fashion.

As a result, we have to adapt the PWPF modulator so that it is also dynamically consistent with our requirements and gives a guarantee that the minimal valve opening time is not violated. The simplest solution implements a logic that forces the outputs of the PWPF modulator to always be switched to on for t_{mib} even if the switch off condition was already met before.

First, a rising edge in the thruster output, which means that the modulator switched from off to on-state, is detected by a flag. The on-state cannot be left until the time specified by t_{mib} has passed and the flag is cleared.

This approach, in a sense, also indirectly penalizes this longer firing, as the pre-filter discharges $f(t)$ to a lower value than usual. Effectively, the next on-switching will be slightly delayed. Further investigations, which are not part of this work, might be necessary in order estimate the effect of this penalization.

Further Limitations Similarly to the PWM, the pulse timings of the PWPF modulator are limited. This means in reality we have an actuation frequency f_{act}, which defines the minimally feasible discretization of the pulse intervals. In the PWM this characteristic was inherently resolved by f_{PWM}.

In addition, currently a continuous realization of the pre-filter for each thruster is required. A conversion to a discrete version of the filter has to be done by transferring into z-space.

Here, the bilinear approximation of Tustin, which transfers a continuous system according to relation

$$s = \frac{2}{t_{act}} \cdot \frac{z-1}{z+1} \tag{2.65}$$

with $t_{act} = \frac{1}{f_{act}}$, is used. Though, any arbitrary other transformation method is also valid. Figure 2.8 shows the adapted modulator.

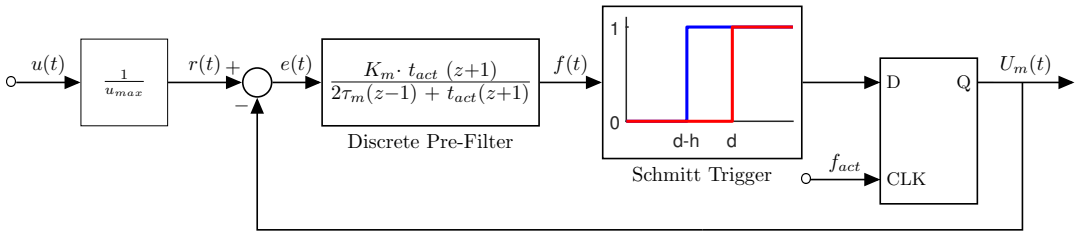


Figure 2.8: Block diagram of the adapted PWPF modulator (adapted from [38])

As an input, we get the cyclically updated thruster assignments $u_i(t)$, which are scaled by u_{max} in order to limit $r(t)$ to 0...1. The discretized PWPF modulator gives a binary output $U_m(t)$ according to the previous definitions. Note that the interlocking logic of the dynamic PWPF modulator is not included in the figure.

2.4.4 Sigma-Delta Modulation

In [41], the applicability of using a first-order Sigma-Delta Modulator (SDM) for spacecraft thruster control is discussed. The SDM has several advantages over the previously shown regular signal modulators. Thus, it is preferred in many technical areas like analog-to-digital filters or audio applications with some of its advantages being the possibility of noise shaping and small-signal actuation.

Note that there exist different adaptions depending on the application and that the depicted Figure 2.9 only shows a popular version, deemed most suited for the application with thrusters.

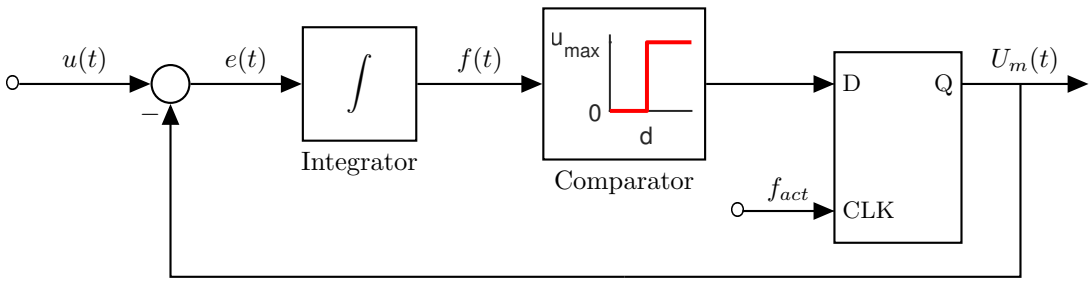


Figure 2.9: Block diagram of a general Sigma-Delta modulator (adapted from [41])

We can see a resemblance with the PWPF modulator and that the SDM can also be summarized by a feedback loop. We use a simplified pre-filter and a comparator instead of a Schmitt Trigger. In our scheme the input once again corresponds to a thruster assignment which is refreshed cyclically due to the control frequency.

The output is given by a two-valued bitstream and on average has to be equal to the input value. This is enforced by the negative coupling. The error of input and the, to an analog value converted, output is first integrated. This integrated signal is fed into a comparator with an arbitrary comparative value. It can be shown that the exact value of the switching condition does not have an influence with the exception of the initial start. The comparator output is switched to 1 if the condition is met. Its value is forwarded to the modulator output depending on the internal clock. A higher frequency of the internal clock leads to a lower error of the average bitstream value, compared to the input.

Adaption to Thruster Application The lengths of the pulses, generated by the general SDM and depicted by the block diagram 2.9, are directly dependent on f_{act}. In the literature, this frequency is also often referred to as the oversampling rate. The SDM output will consist of a lot of very small pulses if kept unaddressed. As the bitstream, given by $U_m(t)$, should only consist of pulses longer than t_{mib}, we have to adapt this SDM version. In the PWPF modulator, mainly the pre-filter with an additional logical interlocking was used to achieve this. Here, we can simply specify

$$f_{act} = \frac{1}{t_{mib}} \tag{2.66}$$

to ensure that no pulse is shorter than t_{mib} even for non-constant $u(t)$.

However, this does not only result in a large discretization error but also enforces pulse time intervals which are multiples of t_{mib}. If the input is a multiple of f_{act} and static, we will get an uniform non-changing pulse sequence. For example, an assignment of $u(t) = \frac{u_{max}}{2}$ will lead to a constant pulse time interval of $2 \cdot t_{mib}$. In general though, we will obtain a pulse sequence with changing pulse frequencies. The average output value is only equal to the input for long observation times. The pulse frequencies are high compared to other modulators, as pre-dominantly pulses with length t_{mib} will be generated. For reasonably high u_i, each thruster will fire multiple times per control time interval, which will increase the wear of the valves.

Alternative Adaption From another point of view, we can just observe the SDM as a special case of the PWPF modulator. We replace the comparator again with a Schmitt Trigger, but keep the general integrator instead of the pre-filter.

Thus, for static input r_c not only a pseudo-linear behavior is obtained but a linear one. Similar to the PWPF modulator one can calculate characteristic formulas to describe the switching properties

$$\begin{aligned} t_{off} &= \frac{t_{mib}}{r_c} \\ t_{on} &= \frac{t_{mib}}{1-r} \\ t_{pulse} &= \frac{1}{t_{mib}} \cdot r(r-1) \\ MF &= r \end{aligned} \tag{2.67}$$

by using $d = t_{mib}$, $d - h = 0$ and assuming an infinite value for f_{act}. Note that this solution again requires an interlocking logic for the dynamic case.

2.5 Comparison of Static Input-Output Characteristics

In this section we want to obtain some impressions about the static input-output behavior of the thruster actuation as a whole block during one control time interval. The goal is to be able to compare different choices for the thruster assignment and modulation by using multiple performance parameters without requiring a closed loop. In practice, not only the wrench error per control time interval and the fuel consumption of an assignment is of interest but also, for example, how pure forces and torques are actuated. In this context a pure or also clean force is obtained if the torque part in the wrench is equal to zero.

For synchronized time intervals of the control and the thrusters, as it is the case for a PWM, it is sufficient to look at the open loop behavior to already receive meaningful information on how the closed-loop system will behave. This requirement is not met by both the PWPF modulator and the SDM. Thus, the main attention of this section is put on combinations of the proposed assignment approaches with a PWM.

All assignments are generated by using functions in MATLAB.

2.5.1 Specification of Different Scenarios and Test Generation

We will randomly draw a high number of sample control wrenches, which have to be located inside the physical controllability envelope. One can quickly identify these wrenches by calculating the quadratic reconstruction error, as it is given by the QP-formulation 2.39. If it is zero, we can guarantee that a theoretical continuous assignment that perfectly realizes the control wrench could be found.

Overall, we will specify four different scenarios. The first scenario samples completely random from the complete controllability envelope, while the second scenario only includes very small wrenches. In the third and forth approach, we will constrain the wrench space to a sphere of exclusive forces and torques respectively.

Some of the previously presented assignment methods will be paired with a PWM. A short list of the investigated assignment methods is given in the following:

1. LP1: Optimization solution for Vanilla LP formulation (equation 2.12); realization with linprog-function of MATLAB with the Dual-Simplex algorithm.
2. LP2: LP formulation with rounding (equation 2.13c)
3. LP3: Expansion of LP assignment with counter-thrust notation (see equation 2.36)
4. MILP1: Optimization solution for standard MILP formulation (presented in Subsection 2.3.3); realization with intlinprog-function of MATLAB; the standard optimization options are selected, except from the cut generation, which is 'intermediate'
5. MILP2: MILP with convergence band in thrust space depending on the discretization (according to equation 2.33)
6. MILP3: MILP with tolerated wrench deviation in wrench space by specifying a bound in equation 2.19; the assignment is allowed to realize forces/torques, which have an absolute value of up to 0.1N/0.1Nm less
7. QP1: Quadratic programming formulation with $\mathbf{k} = 0.01 \cdot \mathbf{1}$ (equation 2.41); using quadprog-function in Matlab with the interior-point-convex algorithm
8. QP2: Quadratic programming formulation with $\mathbf{k} = 0.001 \cdot \mathbf{1}$ (equation 2.41)
9. Pseudoinverse with shift (PI1): Assignment according to equation 2.9 with a null space vector, which is defined by equation 2.10
10. Pseudoinverse with factorization (PI2): Assignment according to equation 2.11
11. Iterative Pseudoinverse (IPI): Scheme presented in Algorithm 1

All optimization algorithms have a specified constraint tolerance of 10^{-8} and an optimality tolerance of 10^{-9}. All assignments will be rounded towards the sixth position after decimal point. The performance will be assessed by the average error in the applied wrench both before and after the modulation, the average fuel consumption of the assignments, the computation time and other parameters.

The influence of different f_{ctrl} and f_{pwm} is also explicitly considered. We are using thrusters with $t_{mib} = 0.015s$ and $u_{max} = 10N$. The control frequencies are chosen from the set $\frac{1}{t_{mib}} \cdot [60\ 10\ 10\ 5]^{-1}\ Hz$, while the PWM frequencies are selected from $\frac{1}{t_{mib}} \cdot [5\ 500]\ Hz$.

2.5.2 Reference Thruster Configuration

The following results are based on a highly redundant reference configuration with overall 24 thrusters. As a special property, this configuration is also symmetric. We recall that for a symmetric thruster configuration no effective wrench is applied if all thrusters fire. Additionally, we also have geometrical symmetry with respect to the planes spanned by the Cartesian axes.

As depicted in Figure 2.10, three thrusters are placed (red pyramids) in each corner of a unit cube (side length $2d = 1m$).

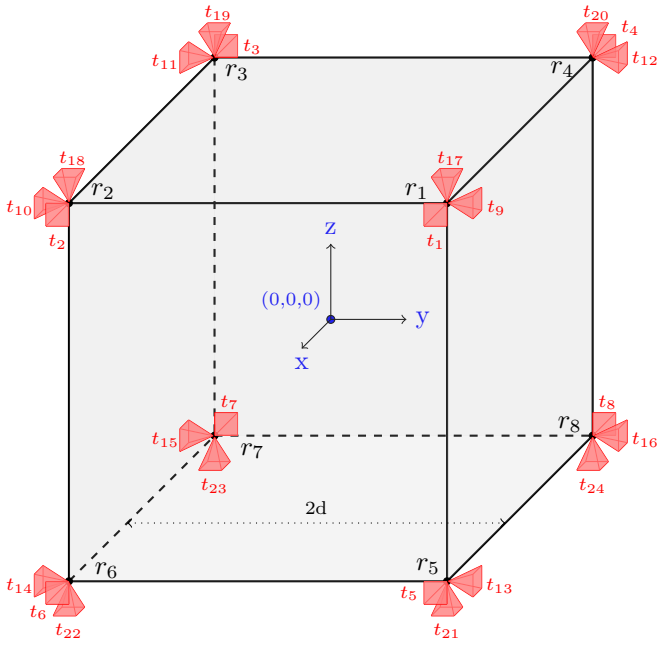


Figure 2.10: Reference Thruster Configuration with 24 thrusters

Each thruster only manipulates three entries of the wrench. The firing direction is indicated by the base of the pyramid. The origin of the geometrical frame is located in the center of the cube.

The numbering of the thrusters can be done artificially. In our case, the numbering derives from the construction approach, which is derived in 3.5. The resulting mapping matrix is given in the following by

$$\mathbf{B}_1(1:12) = \begin{bmatrix} -1 & -1 & 1 & 1 & -1 & -1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 1 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & d & -d & -d & d \\ -d & -d & d & d & d & d & -d & -d & 0 & 0 & 0 & 0 \\ d & -d & d & -d & d & -d & d & -d & -d & d & -d & d \end{bmatrix}$$

$$\mathbf{B}_1(13:24) = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 1 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & -1 & -1 & -1 & 1 & 1 & 1 & 1 \\ -d & d & d & -d & -d & d & d & -d & d & -d & -d & d \\ 0 & 0 & 0 & 0 & d & d & -d & -d & -d & -d & d & d \\ -d & d & -d & d & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

with $d = 0.5m$.

As this section focuses on the effect of the different assignment methods, at this point we will not further investigate if it is possible to realize such a configuration. We solely use it due to the intuitive placement of the thrusters and nice properties of the resulting configuration.

2.5.3 First Scenario: Random Wrench Sampling

In this scenario a total of 150,000 test wrenches are drawn uniformly from the force $[-40N, ..., 40N]$ and the torque set $[-40Nm, ..., 40Nm]$. Each entry of the wrench is drawn independently.

For our configuration, this corresponds to the maximally applicable forces/torques in the different dimensions. The sampled control wrench is inside the controllability envelope and can be realized if the quadratic reconstruction error is zero. Wrenches with non-zero reconstruction error are dropped.

Absolute Wrench Error

First, we want to investigate the wrench error which occurs due to the different assignment methods. We will differentiate not only between force and torque errors but also compare the error after the assignment with the one after the modulation.

For each sample we take the absolute value of the wrench error and sum up the three force entries and the three torque entries. The average error value for all samples is shown in the following tables. This approach limits possible statements to some degree, as we cannot give a statement about the specific force and torque entries which are more prone to errors. On the other hand a quick statement of the total absolute error in the force and torque which is made on average can be given.

Average of Sum of Absolute Force Errors Table 2.1 shows the force errors which occur on average for a random sample during the assignment.

The reasons are infeasible assignments; at least one assigned thrust is not within the set $[0, u_{dead}...u_{max}]$. At this point, the PWM frequency has only an impact for methods that directly include it in their formulation (namely LP3 and MILP2).

In all optimization methods only the cutting of assignments below the dead-zone is relevant and leads to a force error. With rising control frequencies, u_{dead} gets bigger as well. Consequently, it becomes more probable for a random sample to result in an assignment that has to be cleared up. Therefore, we have, in general, larger errors for faster control frequencies. Comparing LP1 and LP2, we can see that by rounding at one half of u_{dead} the average error due to the assignment is also halved. LP3 translates all infeasible assignments to the right hand side of u_{dead} by using counter-thrust. An error can only be observed due to the limited PWM resolution, as assignments below u_{disc} are not corrected.

$f_{ctrl}[Hz]$	**1.11**		**3.33**		**6.67**		**13.33**	
$f_{pwm}[Hz]$	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**
LP1	0.008	0.008	0.074	0.074	0.287	0.287	1.108	1.108
LP2	0.004	0.004	0.037	0.037	0.144	0.144	0.553	0.553
LP3	0.000	0.000	0.003	0.000	0.012	0.000	0.048	0.000
MILP1	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
MILP2	0.034	0.000	0.107	0.001	0.223	0.002	0.496	0.004
MILP3	0.245	0.245	0.246	0.246	0.246	0.246	0.246	0.246
QP1	0.056	0.056	0.227	0.227	0.657	0.657	2.015	2.015
QP2	0.039	0.039	0.224	0.224	0.664	0.664	1.916	1.916
PI1	4.120	4.120	4.118	4.118	4.165	4.165	4.833	4.833
PI2	2.701	2.701	2.796	2.796	3.100	3.100	4.259	4.259
IPI	0.084	0.084	0.169	0.169	0.464	0.464	1.625	1.625

Table 2.1: Scenario 1: Sum of absolute force errors after assignment [N]

We see that the assignments from the MILP1 method are nearly optimal and the wrench error is exclusively coming from the PWM. MILP2 tolerates assignments below u_{disc}. The different algorithm compared to the Simplex seems to prefer these assignments, which results in a larger error compared to the similar LP3. In MILP3, a force deviation of up to $0.1N$ in each dimension was explicitly tolerated if it reduces the amount of consumed fuel. Thus, it comes to no surprise that we have a relatively large deviation, although an absolute lower force value seems to not always be the most fuel conservative solution. Otherwise, we would obtain an error of 0.3N.

In the QP methods, a force deviation is accepted if it reduces the fuel consumption, resulting in a lower overall cost. The prioritization is done by changing the weight parameter. Consequently QP2 has a lower error compared to QP1.

The PI methods have a large deviation, as there is no optimization for different wrenches but only a fixed operation. We can see that PI2 seems to use a more universal nullspace vector than PI1 for the selected configuration. The iterative approach results in a considerable improvement, but it still does not reach the results of the optimization methods.

After the modulation, we will have an additional error, as visible in Table 2.2. We can observe an increase of the average error for all methods. For a small number of discretization steps, resulting from a low value for f_{pwm}, this error becomes quite significant, especially due to the used floor-operation. For a PWM, the ratio of PWM and control frequency is the main tuning parameter to reduce this modulation error.

Operation-wise LP3 and MILP1 are equal in their result and have the lowest deviations. For small control frequencies LP1, LP2 and MILP2 are close, but they fall off for larger frequencies. The errors for the PI methods only marginally increase with the addition of modulation errors, but are still by far the worst in performance. IPI is still worse than the optimization, but close especially for large f_{ctrl}.

$f_{ctrl}[Hz]$	**1.11**		**3.33**		**6.67**		**13.33**	
$f_{pwm}[Hz]$	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**
LP1	0.072	0.009	0.253	0.076	0.611	0.290	1.622	1.113
LP2	0.068	0.005	0.217	0.039	0.472	0.147	1.084	0.558
LP3	0.066	0.001	0.198	0.002	0.397	0.004	0.792	0.008
MILP1	0.066	0.001	0.201	0.002	0.405	0.004	0.826	0.008
MILP2	0.085	0.001	0.255	0.003	0.512	0.005	1.039	0.011
MILP3	0.297	0.246	0.406	0.248	0.587	0.249	0.994	0.253
QP1	0.135	0.056	0.436	0.229	1.014	0.661	2.534	2.019
QP2	0.119	0.039	0.429	0.226	1.006	0.667	2.430	1.920
PI1	4.117	4.120	4.108	4.118	4.123	4.165	4.633	4.831
PI2	2.784	2.702	3.023	2.798	3.499	3.104	4.879	4.264
IPI	0.168	0.085	0.402	0.171	0.875	0.468	2.246	1.631

Table 2.2: Scenario 1: Sum of absolute force errors after modulation [N]

Average of Sum of Absolute Torque Error The same inspection can be done with respect to the torque errors.

Overall, the same observations can be made but only with smaller absolute values. Even though two torque entries are affected by a corrected thrust, on average the torque errors are lower in value. This can be mainly traced back to the lever arm of $0.5m$. The corresponding tables can be found in Appendix B.

Average Absolute Wrench Error in Percent We want to relate the relative error values after the modulation to the demanded force/torque part of the control wrench. Therefore, we will first calculate the absolute error independently for each wrench entry. Then, we compute the relative error.

We differentiate between relative torque and force errors and save them in different destinations. Each sample gives us three relative force and torque errors. Therefore, we can compute average relative errors for a total of 450,000 values for both torques and forces. Note that this approach slightly differs compared to before, where we operated on the sums of the errors of the three distinctive entries for forces and torques in each wrench.

The tables 2.3 and 2.4 show the result of the presented approach in percent. The values for $f_{pwm} = 33.3kHz$ give a good impression about the error coming from the assignment.

We get a very similar ranking of the different methods. The main takeaway from this examination is that the QP and PI formulation have a better performance for this parameter than expected. For the QP approaches, we will generally have larger wrench deviations for larger control wrenches. With respect to the average relative error this results in a better performance value. This is also true for the PI methods.

$f_{ctrl}[Hz]$	**1.11**		**3.33**		**6.67**		**13.33**	
$f_{pwm}[Hz]$	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**
LP1	0.78	0.16	2.90	1.22	6.53	3.58	16.01	11.65
LP2	0.74	0.10	2.28	0.57	5.37	2.06	11.95	6.70
LP3	0.69	0.01	2.01	0.03	4.02	0.04	7.78	0.09
MILP1	0.69	0.01	2.95	0.02	4.41	0.05	8.07	0.10
MILP2	1.03	0.01	4.30	0.03	6.36	0.06	10.43	0.13
MILP3	4.12	4.07	4.31	3.93	5.98	4.13	9.19	3.87
QP1	1.59	0.62	3.39	1.95	7.83	4.93	17.48	13.10
QP2	1.57	0.80	3.76	2.08	7.49	5.21	16.24	12.80
PI1	9.64	9.23	10.46	9.43	11.80	10.14	16.48	14.33
PI2	5.96	5.41	7.58	6.21	10.39	8.32	17.61	14.95
IPI	0.99	0.42	2.56	1.09	5.37	2.99	12.25	9.05

Table 2.3: Scenario 1: Relative average force errors after modulation [%]

$f_{ctrl}[Hz]$	**1.11**		**3.33**		**6.67**		**13.33**	
$f_{pwm}[Hz]$	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**
LP1	1.42	0.13	6.86	3.33	14.57	8.96	33.41	28.03
LP2	1.38	0.07	4.08	2.01	10.75	5.77	20.34	13.18
LP3	1.34	0.01	4.49	0.04	8.01	0.08	16.07	0.15
MILP1	0.95	0.01	2.62	0.03	6.07	0.06	10.92	0.09
MILP2	1.33	0.01	3.12	0.04	6.55	0.07	15.25	0.14
MILP3	5.14	4.45	6.47	4.80	9.74	4.66	13.68	4.61
QP1	1.10	0.31	4.03	1.75	8.47	4.59	19.01	13.60
QP2	1.33	0.68	3.74	2.44	8.62	6.04	20.97	18.45
PI1	11.41	11.16	12.25	11.68	13.69	12.80	17.14	16.16
PI2	8.50	8.02	9.60	8.62	11.60	10.18	16.61	14.68
IPI	0.84	0.31	2.25	1.00	4.79	2.95	10.91	8.55

Table 2.4: Scenario 1: Relative torque errors after modulation [%]

Average standard Deviation of Actuated Wrench Error We want to develop a better understanding of the error distribution. To decrease the number of plots we will only investigate the behavior for $f_{ctrl} = 3.33Hz$.

We recall that this results in $u_{dead} = 0.5N$. Only the sum of absolute force errors will be looked at, but the same behavior can be observed for the torque part of the wrench. In the selected configuration, the thrusters solely fire along one axis. This means that a decoupled force along one of the three axes will be generated by each thruster. This knowledge will be important in the following.

Table 2.5 shows us the Standard Deviation (STD) of the sum of absolute force errors. By assuming a Gaussian distribution, we can specify this kind of error by calculating the mean and the standard deviation. It has to be noted that one could also take each force dimensions specifically and use the signed force errors. Then it suffices to only take the standard deviation, since the signed force errors for most assignment methods are relatively well centered around zero.

$f_{ctrl}[Hz]$	**1.11**		**3.33**		**6.67**		**13.33**	
$f_{pwm}[Hz]$	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**
LP1	0.036	0.030	0.149	0.152	0.375	0.409	0.971	1.074
LP2	0.029	0.015	0.095	0.076	0.213	0.205	0.502	0.535
LP3	0.027	0.000	0.081	0.001	0.160	0.002	0.322	0.003
MILP1	0.027	0.000	0.081	0.001	0.163	0.002	0.332	0.003
MILP2	0.044	0.000	0.131	0.001	0.258	0.003	0.528	0.005
MILP3	0.078	0.067	0.131	0.067	0.218	0.067	0.385	0.067
QP1	0.084	0.071	0.306	0.299	0.715	0.725	1.638	1.670
QP2	0.084	0.075	0.305	0.300	0.682	0.689	1.512	1.507
PI1	3.847	3.883	3.745	3.837	3.506	3.647	2.821	3.014
PI2	2.259	2.273	2.215	2.250	2.176	2.228	2.343	2.417
IPI2	0.381	0.388	0.406	0.414	0.594	0.606	1.347	1.401

Table 2.5: Scenario 1: Standard deviation of sum of absolute force errors after modulation [N]

MILP1 and LP3 not only have a lower mean error, but also are more sharp, visible by a low STD value. The rounding in LP2 heavily narrows the error range which is obtained by LP1. QP1 and QP2 also have larger deviations between the errors. Once again, PI1 and PI2 perform the worst, as they not only have a larger mean error, but also a large variance. This is also visualized by the following histograms.

Figure 2.11 illustrates the distribution of errors for the LP methods. For high PWM frequencies, the main error comes from the assignment. Only a small fraction of the samples will result in a large error, as visible on the right side plot. This corresponds to the previous results.

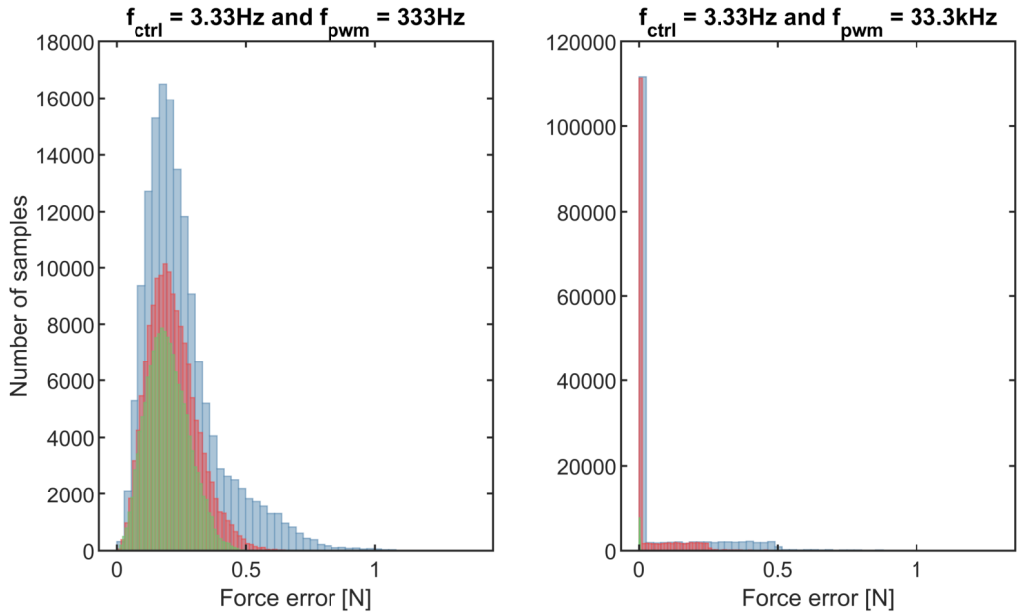


Figure 2.11: Scenario 1: Distribution of sum of absolute force errors for LP1 (blue), LP2 (red) and LP3 (green)

Due to the selection of $f_{ctrl} = 3.33Hz$, we have a reasonably low u_{dead}. The LP methods will have the same performance as the MILP ones if there is no assignment inside the dead-zone. However, there exist control wrenches for which LP1 assigns thrust within the dead-zone. We know that $u_{dead} = 0.5N$ and that each force direction is actuated by different thrusters. Therefore, this correction should very rarely occur. The plot on the right shows that we have most of the times only small corrections for only a few thrusters. In theory though, there might exist wrenches which result in a lot of assignments close to u_{dead} and one could try to construct a control wrenches which result in maximal errors. Due to the rounding, the maximum possible error is halved in LP2. For sufficiently high f_{pwm}, LP3 has a very sharp distribution.

The left plot shows how the distribution is affected for a lower f_{pwm}. This selection of parameters results in $u_{disc} = 0.1N$. Thus, we know that the absolute modulation error is bounded by $0.1N$ for each thruster and only occurs, if we are above u_{dead}. For all thrusters we can observe a combination of modulation and assignment errors. LP1 has a significantly larger spread of its errors than LP2 and LP3. This can be traced back to the cutting off of assignments, which is reduced in LP2 and removed in LP3.

The results in Figure 2.12 show how the QP optimization methods are distributed. We would expect a similar result as obtained for LP1, as these methods also do not have a knowledge about u_{dead}. However, we get a worse overall performance compared to LP1. We recall that we can have infinitely many solutions for the assignment problem. The QP algorithm more often assigns low valued thrusts, which then are more often also below u_{dead}. The different k-penalties (compare QP1 and QP2) have only a marginal effect here.

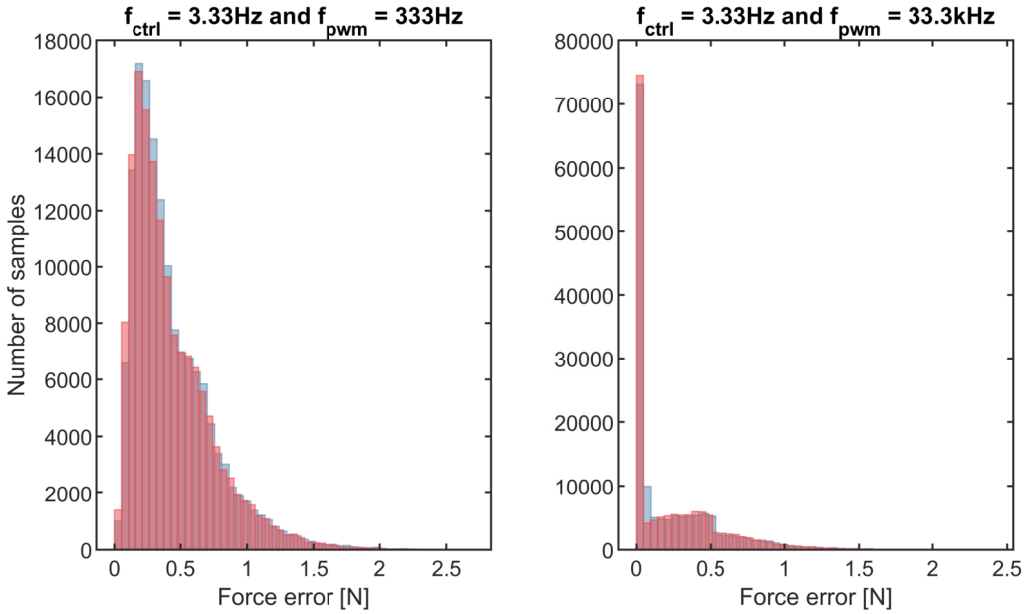


Figure 2.12: Scenario 1: Distribution of sum of absolute force errors for QP1 (blue) and QP2 (red)

The error of the MILP methods, depicted in Figure 2.13, is only affected by f_{pwm} and the resulting discretization resolution of the assigned thrusts. On the right side we have the results for a very high discretization $u_{disc} = 0.001N$. The spread of the errors is at around $4 \cdot u_{disc}$ for MILP1. This corresponds to four thrusters, which had their assignment maximally corrected by the floor-operation in the modulation. MILP2 has a larger error than MILP1, as also assignments $u_i < u_{disc}$ are tolerated and thus add to the discretization errors. MILP3 is not depicted on the right plot, as the included force deviation boundaries of $0.1N$ result in summed deviations of up to $0.3N$. For high discretizations, the errors are located around the values $0.1N$, $0.2N$ and $0.3N$. The left side plot has effectively the same distribution, but only with $u_{disc} = 0.1N$.

Figure 2.14 depicts the results for PI1, PI2 and IPI. For both PI1 and PI2, the resulting error of some control wrenches can be quite significant. IPI is closer to the optimization results. However this method still has the problem that the performance heavily degrades for a small subset of control wrenches.

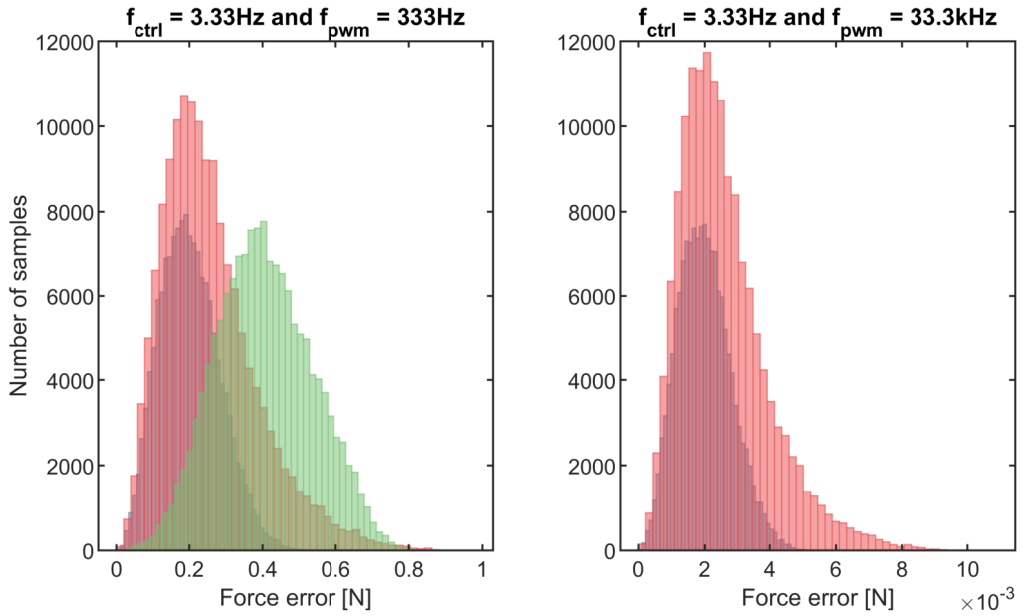


Figure 2.13: Scenario 1: Distribution of sum of absolute force errors for MILP1 (blue), MILP2 (red) and MILP3 (green); right plot without MILP3

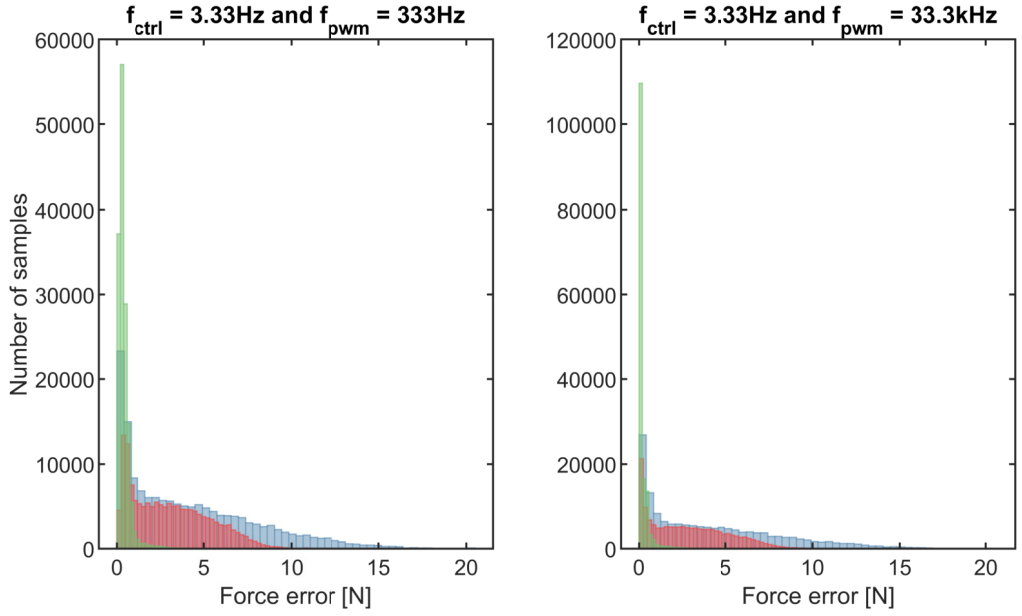


Figure 2.14: Scenario 1: Distribution of sum of absolute force errors for PI1 (blue), PI2 (red) and IPI (green)

Amount of Corrected Thrust on Average after Assignment Step

We can approximate the effect of u_{dead} and the resulting assignment error by summing up the amount of corrected thrust for every $u_i < u_{dead}$ for each assignment (see Table 2.6). Obviously the methods, where this parameter is directly included in the optimization (MILP1, MILP3) or where the correction is part of the assignment (LP2), have zero corrected thrust.

In LP3 and MILP2, we tolerate assignments $u_i < u_{disc}$, so that the amount of corrected thrust is depending on this value. We can also verify a statement from the previous section. Implicitly, the Simplex algorithm prefers sparse assignments, resulting in less thrust corrections compared to the QP-methods but also MILP2. This can be also seen when we later look at the average number of assigned thrusters. Due to this fact also LP3 is far superior to MILP2.

The PI methods once again are not well-suited, while IPI is surprisingly well-off. This has to do with the fact that during each iteration we assign the maximum thrust to some thrusters and assign lower values only during the last step.

$f_{ctrl}[Hz]$	**1.11**		**3.33**		**6.67**		**13.33**	
$f_{pwm}[Hz]$	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**
LP1	0.008	0.008	0.075	0.075	0.299	0.299	1.206	1.206
LP2	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
LP3	0.000	0.000	0.003	0.000	0.012	0.000	0.049	0.000
MILP1	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
MILP2	0.039	0.000	0.118	0.001	0.251	0.002	0.567	0.004
MILP3	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
QP1	0.035	0.035	0.223	0.223	0.716	0.716	2.339	2.339
QP2	0.038	0.038	0.238	0.238	0.734	0.734	2.213	2.213
PI1	0.012	0.012	0.118	0.118	0.553	0.553	2.829	2.829
PI2	0.016	0.016	0.151	0.151	0.597	0.597	2.399	2.399
IPI	0.009	0.009	0.099	0.099	0.428	0.428	1.867	1.867

Table 2.6: Scenario 1: Amount of corrected thrust after the assignment [N]

Average Overactuation

The nondeterministic correction of thrust assignments might result in overactuation of some dimensions of the control wrench. In this case, the absolute values of the actuated forces and torques in the wrench exceed the demanded ones. For the notion of overactuation, we also included if the sign of the actuated wrench parts is different in comparison to their corresponding control wrench entries.

This behavior is highly undesired since it might cause problems with the stability of the control. While most controls can handle too small actuation relatively well, a too large actuation or even actuation in the opposite direction is more problematic. Especially, if one has the final limit cycle in mind (see Chapter 4).

Thus, the formulation used in MILP3 was not only motivated by allowing the optimizer to find slightly deviating wrenches that are more fuel-conservative, but partly also as a solution to this problem. We can see in the tables 2.7 and 2.8 that this method fares the best with respect to overactuation. Overactuation is also reduced in MILP1, MILP2 and LP3, as all include notions about u_{dead}.

$f_{ctrl}[Hz]$	**1.11**		**3.33**		**6.67**		**13.33**	
$f_{pwm}[Hz]$	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**
LP1	0.015	0.003	0.060	0.031	0.167	0.122	0.505	0.464
LP2	0.014	0.002	0.049	0.019	0.123	0.075	0.351	0.298
LP3	0.012	0.000	0.038	0.000	0.076	0.001	0.152	0.001
MILP1	0.010	0.000	0.029	0.000	0.057	0.001	0.111	0.001
MILP2	0.016	0.000	0.051	0.000	0.104	0.001	0.221	0.002
MILP3	0.003	0.001	0.009	0.001	0.026	0.001	0.075	0.001
QP1	0.021	0.016	0.098	0.095	0.273	0.296	0.793	0.920
QP2	0.023	0.017	0.104	0.104	0.279	0.309	0.736	0.867
PI1	0.014	0.005	0.072	0.052	0.265	0.259	1.274	1.433
PI2	0.010	0.006	0.053	0.051	0.168	0.189	0.561	0.668
IPI	0.030	0.021	0.080	0.059	0.207	0.191	0.648	0.695

Table 2.7: Scenario 1: Average sum of overacted forces after the modulation [N]

$f_{ctrl}[Hz]$	**1.11**		**3.33**		**6.67**		**13.33**	
$f_{pwm}[Hz]$	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**
LP1	0.014	0.004	0.056	0.033	0.149	0.120	0.431	0.426
LP2	0.013	0.002	0.043	0.018	0.103	0.069	0.276	0.255
LP3	0.012	0.000	0.035	0.000	0.069	0.001	0.139	0.001
MILP1	0.009	0.000	0.025	0.000	0.049	0.000	0.093	0.001
MILP2	0.015	0.000	0.044	0.000	0.090	0.001	0.182	0.002
MILP3	0.002	0.001	0.006	0.001	0.015	0.001	0.049	0.001
QP1	0.018	0.015	0.085	0.090	0.235	0.271	0.647	0.791
QP2	0.021	0.018	0.097	0.104	0.252	0.297	0.631	0.787
PI1	0.006	0.004	0.038	0.038	0.148	0.168	0.661	0.782
PI2	0.005	0.003	0.025	0.029	0.084	0.107	0.291	0.374
IPI	0.016	0.008	0.061	0.050	0.175	0.181	0.518	0.598

Table 2.8: Scenario 1: Average sum of overacted torques after the modulation [Nm]

The other methods perform worse. Furthermore a higher f_{pwm} can only improve the overactuation for small f_{ctrl}. For larger f_{ctrl} most of the overactuation results from the correction of dead-zone assignments and cannot be recovered by a better thrust discretization.

If low overactuation is an important requirement for the thruster actuation, MILP3 could be adapted even more. Alternatively, one could try out to artificially lower the control wrench before the assignment step to get the same result. Then, an unforeseen correction of dead-zone assignments or modulation errors will not by accident overactuate the system in some directions.

Average Fuel Consumption

All optimization methods optimize for the fuel consumption. The differences mainly come from the correction of dead-zone assignments. This means that another wrench is actuated which, however, is more fuel conservative.

Still, three main aspects can be brought up and backed up by the results in Table 2.9.

$f_{ctrl}[Hz]$	**1.11**		**3.33**		**6.67**		**13.33**	
$f_{pwm}[Hz]$	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**
LP1	68.42	68.52	68.17	68.45	67.69	68.23	66.36	67.31
LP2	68.43	68.53	68.25	68.53	67.99	68.52	67.58	68.53
LP3	68.46	68.56	68.47	68.83	68.89	69.72	71.18	73.33
MILP1	68.44	68.53	68.27	68.57	68.10	68.70	68.11	69.31
MILP2	68.39	68.53	68.13	68.57	67.76	68.70	67.13	69.30
MILP3	67.87	67.97	67.71	68.00	67.53	68.12	67.50	68.69
QP1	68.26	68.42	67.78	68.23	66.89	67.73	64.62	66.11
QP2	68.32	68.48	67.83	68.28	66.94	67.78	64.80	66.30
PI1	136.69	137.01	135.94	136.90	134.59	136.46	130.74	134.16
PI2	62.46	62.63	62.01	62.50	61.13	62.04	58.63	60.23
IPI	69.36	69.52	68.98	69.42	68.24	69.09	66.13	67.65

Table 2.9: Scenario 1: Average fuel consumption [N] (assumed proportional to sum of assignments)

First, an allowed wrench deviation will lead to a reduced fuel consumption. We can see this fact when we compare MILP1 and MILP3 or alternatively the results for both QP methods.

Second, the assignments due to PI1 and PI2 are not fuel optimized and will waste a lot of it, while still resulting in large wrench errors.

Third, when we compare the methods with the lowest wrench errors MILP1 and LP3, we can see that MILP1 finds more conservative assignments than the simple feasible one coming from LP3. Depending on the amount of redundancy in the configuration, we will be closer to LP1. f_{ctrl} directly affects the difference between both. If LP1 does not include an assignment inside the dead-zone, there will not be a difference between MILP1, LP1, LP2 and LP3 in the fuel consumption.

Average Number of Used Thrusters

Only in the MILP formulations a direct penalization of the number of used thrusters is included. Thus, it comes to no surprise that these methods use less thrusters on average. Note that in order to derive this parameter, the number of assignments unequal to zero were summed up and averaged over all samples. The results are shown in Table 2.10. The other optimization methods do not differentiate between solutions which have the same fuel consumption but different numbers of active thrusters.

$f_{ctrl}[Hz]$	**1.11**		**3.33**		**6.67**		**13.33**	
$f_{pwm}[Hz]$	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**
LP1	9.86	9.86	9.86	9.86	9.86	9.86	9.86	9.86
LP2	9.81	9.81	9.71	9.71	9.56	9.56	9.26	9.26
LP3	9.94	9.96	10.10	10.16	10.34	10.46	10.82	11.06
MILP1	9.23	9.23	9.24	9.24	9.26	9.26	9.32	9.32
MILP2	10.38	10.39	10.40	10.39	10.46	10.34	10.60	10.43
MILP3	9.15	9.15	9.16	9.16	9.17	9.17	9.23	9.23
QP1	11.96	11.96	11.96	11.96	11.96	11.96	11.96	11.96
QP2	11.97	11.97	11.97	11.97	11.97	11.97	11.97	11.97
PI1	23.00	23.00	23.00	23.00	23.00	23.00	23.00	23.00
PI2	12.00	12.00	12.00	12.00	12.00	12.00	12.00	12.00
IPI2	11.70	11.70	11.70	11.70	11.70	11.70	11.70	11.70

Table 2.10: Scenario 1: Average number of assigned thrusters

Assignments inside the dead-zone get cleared out before the modulation, which is why for larger f_{ctrl}, on average, fewer thrusters are used by the LP methods compared to MILP, as depicted by Table 2.11.

$f_{ctrl}[Hz]$	**1.11**		**3.33**		**6.67**		**13.33**	
$f_{pwm}[Hz]$	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**
LP1	9.76	9.76	9.56	9.56	9.26	9.26	8.65	8.65
LP2	9.81	9.81	9.71	9.71	9.56	9.56	9.26	9.26
LP3	9.92	9.96	10.04	10.16	10.22	10.46	10.58	11.06
MILP1	9.23	9.23	9.24	9.24	9.26	9.26	9.32	9.32
MILP2	9.20	9.23	9.13	9.24	9.04	9.26	8.88	9.31
MILP3	9.15	9.15	9.16	9.16	9.17	9.17	9.23	9.23
QP1	11.48	11.48	10.90	10.90	10.23	10.23	9.14	9.14
QP2	11.48	11.48	10.85	10.85	10.18	10.18	9.17	9.17
PI1	22.86	22.86	22.55	22.55	21.97	21.97	20.48	20.48
PI2	11.80	11.80	11.40	11.40	10.80	10.80	9.60	9.60
IPI	11.59	11.59	11.32	11.32	10.89	10.89	9.93	9.93

Table 2.11: Scenario 1: Average number of actuated thrusters

The solutions within the QP methods and IPI actuate slightly more thrusters than the LP solutions. We can see that PI1 will actuate nearly every thruster; PI2 is better in this regard.

A lower number of used thrusters reduces the wear of the valves and the life-time of the system potentially increases. Assuming the same fuel consumption, assignments with less active thrusters will have a larger assigned value and should be the preferred solutions. Thus, their firing time also increases.

Average Computation Time

As a final performance parameter, we want to investigate how time consuming the calculation of the different methods is (see Table 2.12). Note that even though the simulation was conducted on one machine, also other internal effects might influence the computation time. In addition, a change in the optimization parameters might heavily affect the required time until a solution is obtained. As only the functions available through Matlab were used, no in-depth run-time optimization was done. Thus, this table should be only seen as a very general comparison to get a first impression of the relative run-times.

$f_{ctrl}[Hz]$	**1.11**		**3.33**		**6.67**		**13.33**	
$f_{pwm}[Hz]$	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**
LP1	8.62	8.56	8.55	8.55	8.55	8.55	8.55	8.55
LP2	8.64	8.63	8.63	8.63	8.63	8.63	8.64	8.63
LP3	8.49	8.47	8.47	8.46	8.46	8.46	8.46	8.46
MILP1	39.95	39.95	40.29	40.31	43.42	43.43	48.43	48.45
MILP2	40.52	40.36	40.93	40.69	43.35	44.05	45.34	49.30
MILP3	42.56	42.53	42.62	42.64	46.55	46.57	51.14	51.16
QP1	2.36	2.36	2.36	2.36	2.36	2.36	2.36	2.36
QP2	2.02	2.02	2.02	2.02	2.02	2.02	2.02	2.02
PI1	0.02	0.02	0.02	0.02	0.02	0.02	0.02	0.02
PI2	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
IPI	0.25	0.25	0.25	0.25	0.25	0.25	0.25	0.25

Table 2.12: Scenario 1: Computation time [ms]

As expected the PI approaches are the fastest methods, as only one simple matrix multiplication has to be carried out for each control wrench. IPI iteratively improves the result, but in general also only requires a small amount of iterations before convergence.

Surprisingly the QP methods are significantly faster than the LP ones. The computation time is about four times lower.

Due to their complexity MILP problems will require more resources than regular LP problems. Unfortunately MILP2 proves to be on average not significantly faster than MILP1, as it was observed after initial tests.

All methods are relatively stable with respect to f_{ctrl}, except from MILP. Here, the computation time rises with larger f_{ctrl}.

Robustness and Wrenches outside the controllability envelope

In this context, we use the word robustness to express how difficult the tuning of the methods is. The method is not robust if it often cannot find an assignment, even if there exists one that satisfies the constraints. Alternatively, this concept also denotes if certain adjustments had to be made towards the settings of the solver.

Mainly the MILP methods and partly also the LP ones caused some problems in this regard. With the regular settings of the solver, sometimes no feasible assignment could be found. In both methods some small adaptions had to be made in order to always get a solution if there exists one. For example in the MILP, the cut generation had to be set to a more advanced setting. No problems were observed for the QP and pseudoinverse approaches.

The LP and MILP methods cannot find an assignment, if the control wrench is outside the controllability envelope. For both we have an explicit constraint that enforces a correct assignment with respect to the control wrench. In contrast, the QP method has the reduction of the wrench error directly inside the cost function. Thus, we can immediately obtain the closest possible wrench within the controllability envelope. The length of the control wrench vector is just scaled down accordingly. IPI, PI1 and PI2 have no direct formulation of the wrench error. Therefore, they are also capable to provide assignments for wrenches outside of the controllability envelope.

2.5.4 Second Scenario: Small Wrenches

In the previous scenario it was assumed that there is no clustering of the wrench errors depending on the length and direction of the control wrench vector. However, we saw that the optimization methods are reasonably close in performance for low control frequencies. For larger f_{ctrl} only the MILP and the extended LP3 method can hold the wrench error reasonably low.

It is obvious that especially small control wrenches will result in infeasible assignments for all methods without an included notion of the dead-zone. Note that by application of small perturbations, we could construct a control wrench, which results in at least one infeasible assignment for e.g. LP1 for any control wrench. In this scenario we will limit ourselves to small control wrenches, as they are particularly important for high accuracy control.

We will sample 6000 wrenches, for which the entries are drawn independently from the limited force $[-1N, ..., 1N]$ and the torque set $[-1Nm, ..., 1Nm]$. Here, only a short discussing regarding the main differences compared to the general sample space and the most important results is conducted. The resulting tables are fully included in appendix B.

Keeping the configuration in mind, for example, for LP1 we know that the control wrench $\mathbf{T} = [1\ 0\ 0\ -0.5\ 0.5\ 0]$ will result in an assignment $u_4 = 1N$. This means that this control wrench will result in no actuation for $f_{ctrl} = 13.33Hz$ for both LP1 and LP2, but also the PI and QP methods, as $u_{dead} = 2N$. Only the MILP methods and LP3 have a notion of the dead-zone and will find feasible assignments. For smaller control frequencies with smaller u_{dead}, we can find at least some control wrenches which result in actuation. We can see that the main difference of the MILP and the other approaches can be identified for such small wrenches with entries in the area of u_{dead}.

Another important take away from this scenario is that the computation time of the MILP methods heavily increase compared to before. The MILP2 approach seems to increase the convergence speed for small f_{pwm} but fails to help for high f_{pwm}. In the worst case, an average of over one second of computation time could be observed, which is 25 times longer than before. Also, some samples needed significantly more time. In the MILP approach, we introduced Big-M constraints in order to penalize dead-zone assignments. The goal is to penalize these solutions so that they have a high cost, but they should not violate any constraints. Though, depending on the coefficient u_{num}, we also create infeasible assignment regions, as discussed in more detail in 2.3.3. This results in a highly non-linear solution space, which requires more cuts and a longer computation in the MILP algorithm. With MILP2 we seem to be able to create a larger convergence zone for assignments close to zero, which reduces the computation time, but only if this zone is sufficiently high (f_{pwm} not to high). A more in-depth investigation is necessary to clarify how to resolve this problem and reduce the variance in convergence speed.

2.5.5 Third Scenario: Pure Force Sampling

For some special control types it is necessary to actuate pure forces or torques. We recall that our definition of a pure force meant no torque entry unequal to zero in the control wrench. At first, pure forces with a vector length of $10N$ are sampled. The samples are generated by triangulation of a unit sphere based on icosahedron face partitioning and by scaling to $10N$ length. This allows us to generate control forces from a sphere with the same distance from the origin. We can also visualize certain behaviors if we limit ourselves to a subset of possible control wrenches by only sampling pure forces.

As the output of the tables has no real surprises, when comparing to scenario 1, no discussion of specific performances will be given here. Although, the resulting tables are given in appendix B for reference. At this point we want to focus on the distribution of the fuel consumptions for different three-dimensional force directions. In addition we want to illustrate some of the previously identified properties of the different assignment methods. With the help of the following plots we will get a first insight how the thruster positioning affects the solution space of applicable wrenches and how this solution space changes with respect to the selection of the control frequency and assignment method. The results will also be restated in Chapter 3.

Influence of Control Frequency on Fuel Distribution We saw before that, for sufficiently high f_{pwm} and no uncertainties, the MILP1 method will provide nearly ideal thruster assignments. Minimal deviations and a minimal amount of used thrusters are obtained due to the special formulation of the optimization problem. In a sense, the MILP1 method will provide the user with the best possible assignment for a given geometrical configuration of thrusters.

Ideally, all force directions with the same vector length require the same amount of fuel. We recall that we introduced the fuel consumption as a proportional property to the sum of all thruster assignments. This corresponds to an L1-norm, while the force length is calculated by an L2 norm. We can get an ideal fuel distribution for the one-sided thrusters only for infinitely many thrusters, with firing directions that are distributed equally along a sphere. For only a small number of thrusters, we will get preferred directions of wrenches, which are more conservative than others. Especially when including arbitrary torques, the distribution of fuel consumption in the six-dimensional controllability envelope is very complex.

The figures 2.15 and 2.16 show how fuel conservative different pure control forces are. The fuel consumption for each sampled control wrench is plotted on the left, while the number of required thrusters is shown on the right side. We use $f_{pwm} = 33.3kHz$, which allows us to ignore errors due to the modulation. By reminding ourselves that the configuration was selected in a way that allows to generate decoupled, pure forces by only firing two thruster at a time, it is understandable why the Cartesian axes are the preferred directions for a fixed control torque with a length of $0Nm$. For example, $u_1 = u_6 = 5N$ will result in a pure negative force of $-10N$ in x-direction, which means that we also require only 10 units of fuel. Overall, six optimal directions, where only one force direction has a value unequal to zero, can be identified. A combination of two control forces on the force sphere leads to an increase in fuel, while the most fuel is required if all three directions are included.

Due to the dead-zone, small deviations of the optimal directions can lead to a sudden increase in fuel consumption. The control force can only be actuated with an additional thruster. With an increase of f_{ctrl}, the fuel distribution becomes more non-smooth.

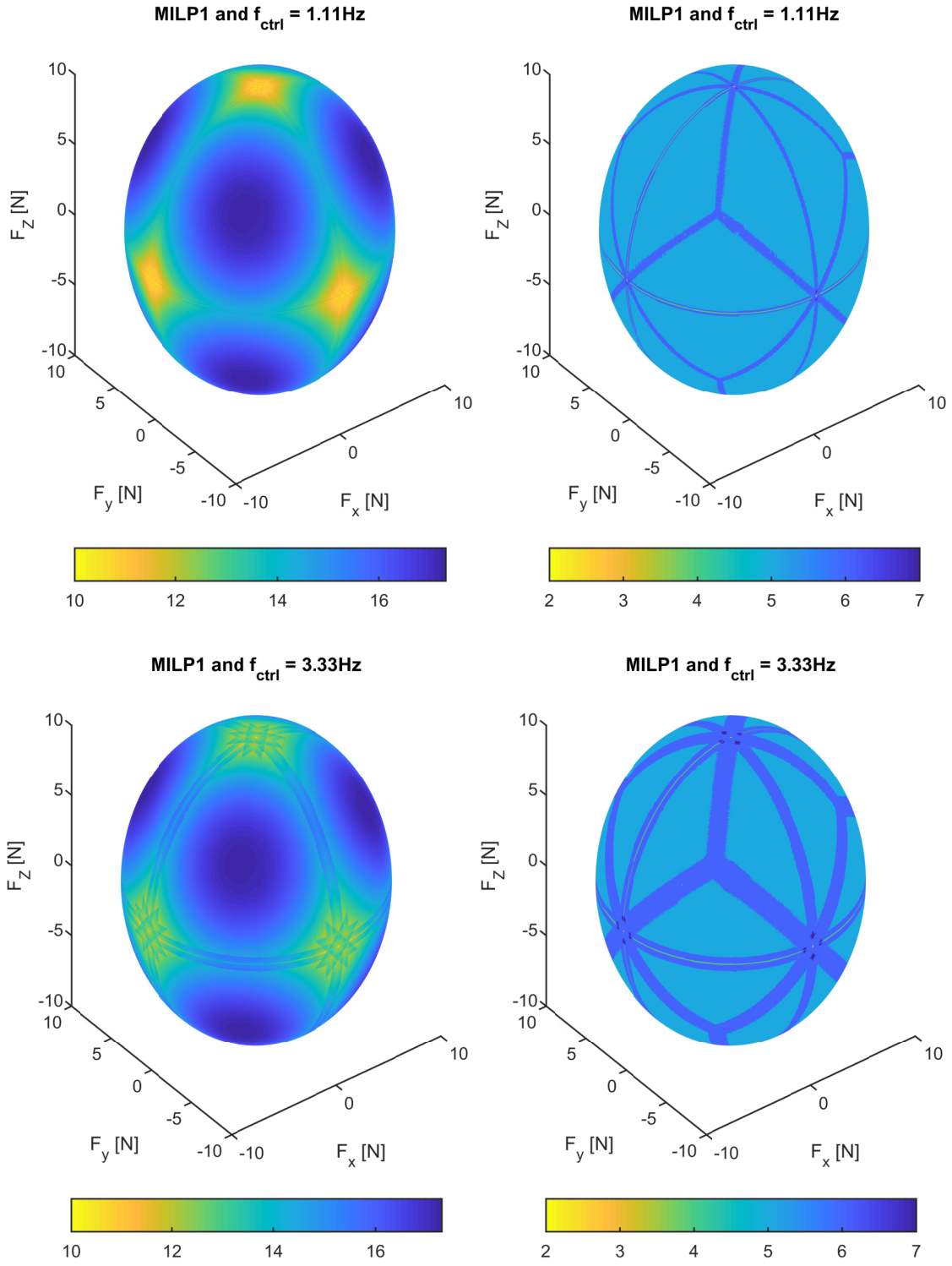


Figure 2.15: Scenario 3: Fuel consumption (left) and required number of thrusters (right) for ideal actuation of the forces on the sphere for MILP1 and $f_{pwm} = 33.3kHz$

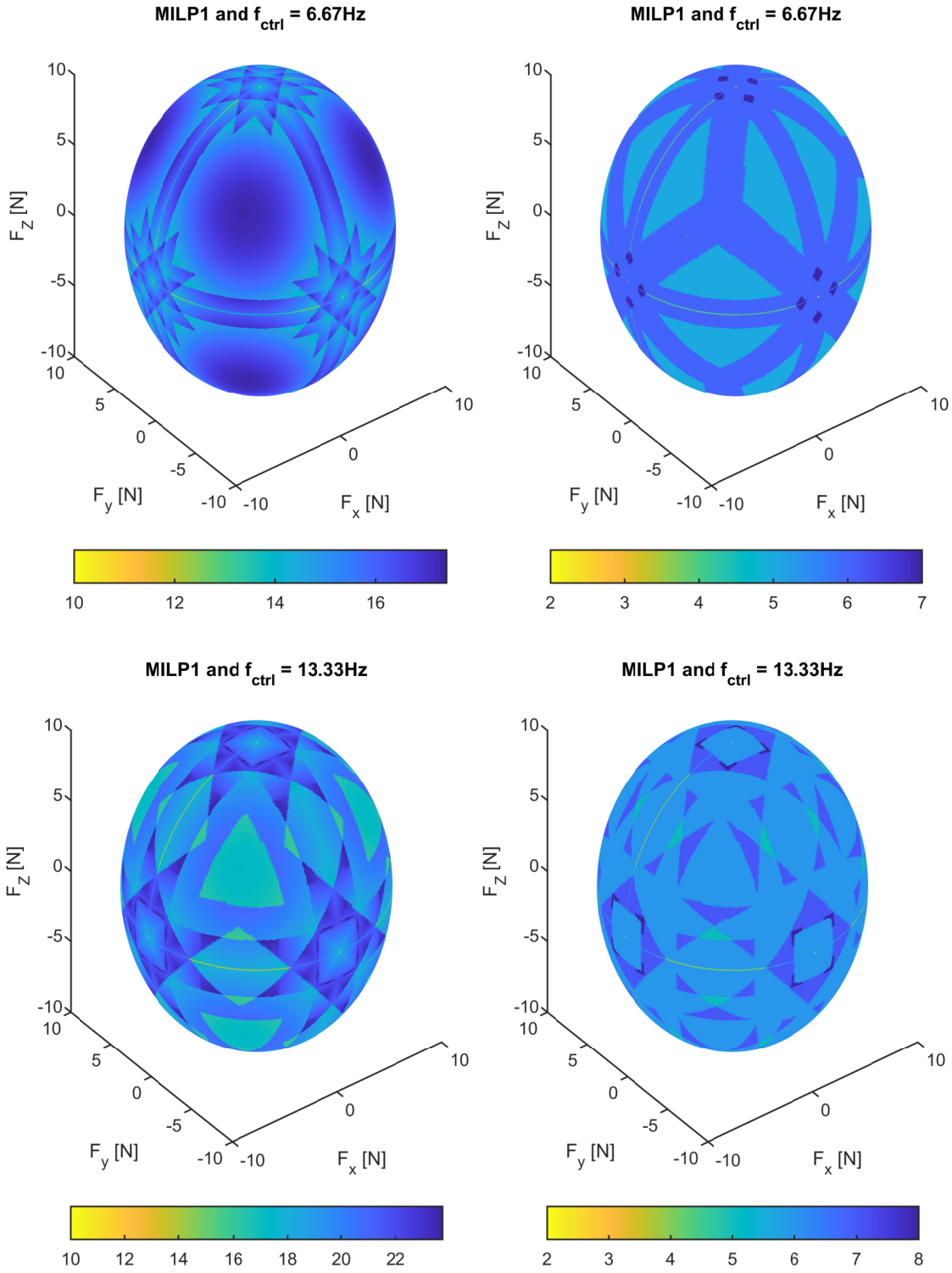


Figure 2.16: Scenario 3: Fuel consumption (left) and required number of thrusters (right) for ideal actuation of the forces on the sphere for MILP1 and $f_{pwm} = 33.3kHz$

Actuated Forces and Fuel Consumption of LP3 Also with the LP3 method, the control wrenches are perfectly actuated for infinite f_{pwm}. However due to the simplified notion of counter-thrust, not the most fuel-conservative solution is found. In Figure 2.17 it can be seen that the sampled forces are still nearly ideally actuated for $f_{pwm} = 33.3kHz$ but require an increased amount of fuel.

We can also see that the distribution of the fuel consumption is different than the one of the MILP1 method and that the output is not symmetrical to the Cartesian axes anymore.

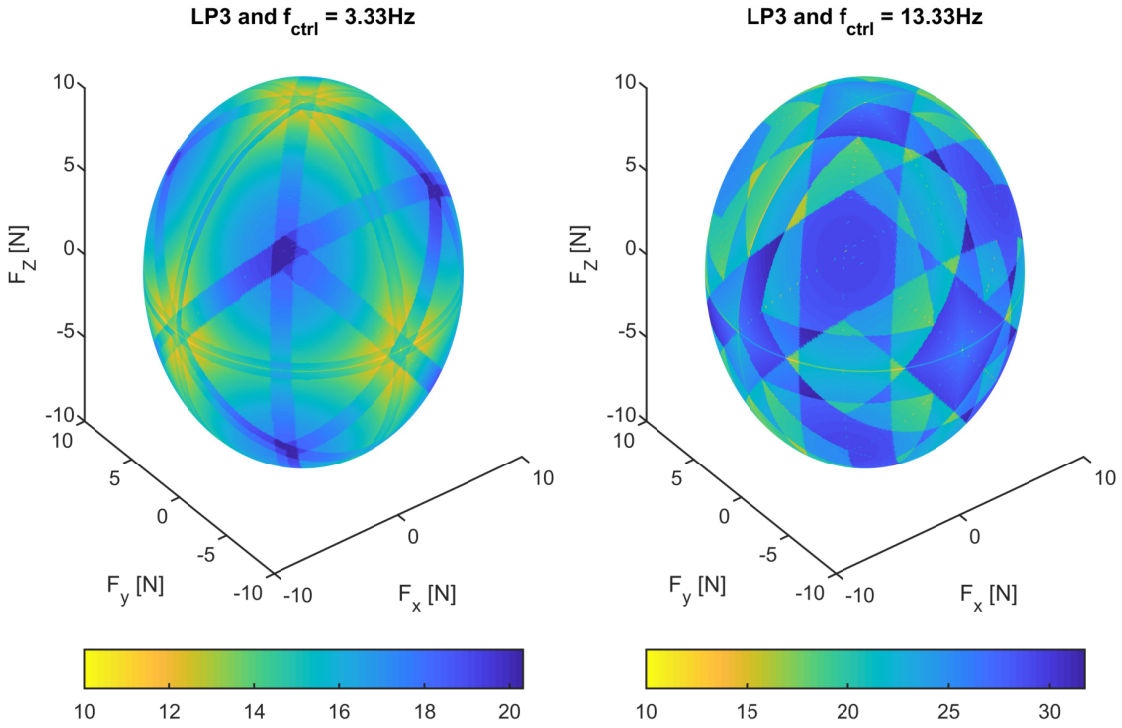


Figure 2.17: Scenario 3: Fuel consumption of actuated forces for two different control frequencies for LP3 and $f_{pwm} = 33.3kHz$

Actuated Forces and Fuel consumption of LP1 Figure 2.18 illustrates how the controllability is reduced when using the LP1 method. Due to cutting of dead-zone assignments, certain forces cannot be actuated. They are mapped to feasible assignments, which results in a force error. For $f_{ctrl} = 3.33Hz$ only some small bands, where additional low-firing thrusters would be required, are affected. The solution space degenerates further, if f_{ctrl} gets larger. The actuated forces can now also be located inside the sphere.

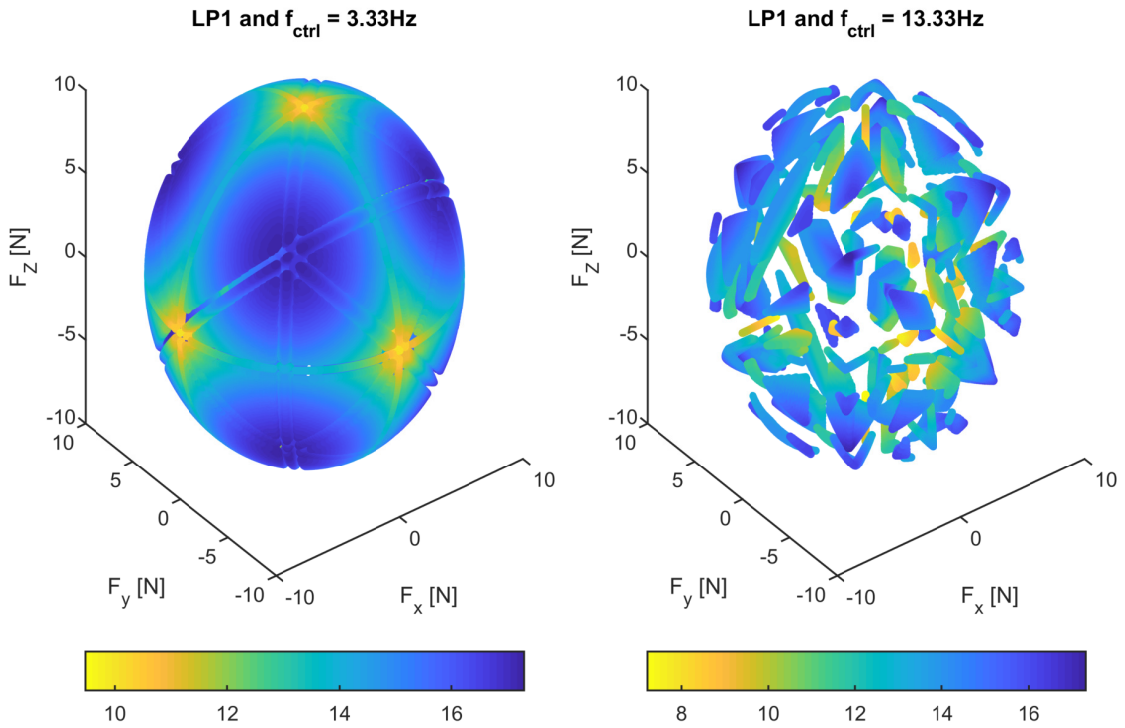


Figure 2.18: Scenario 3: Fuel consumption of actuated forces for two different control frequencies for LP1 and $f_{pwm} = 33.3kHz$

Actuated Forces and Fuel Consumption of QP1 As discussed in scenario 1, on average the QP formulation results in more actuated thrusters with lower assigned thrust values. Thus, the solution space degenerates faster as well. In Figure 2.19 we can see that for $f_{ctrl} = 13.33Hz$ only forces along the Cartesian axes are actuated.

Important Take-Away Due to the configuration, we obtain preferred wrench vectors. Compared to optimization methods with an integrated dead-zone notion, we can see that in the other methods, we not only have significant errors for small wrenches, but certain wrenches around the optimal directions will be erroneous. Each method has a different distribution of "weak" wrenches, which cannot be actuated with high accuracy. This property also changes depending on u_{dead}.

Independent from the assignment method, we have an underlying physical controllability envelope with a distribution of minimal fuel consumption. Even the best assignment method cannot get better than this specification. The assignments from MILP1 come the closest to this distribution.

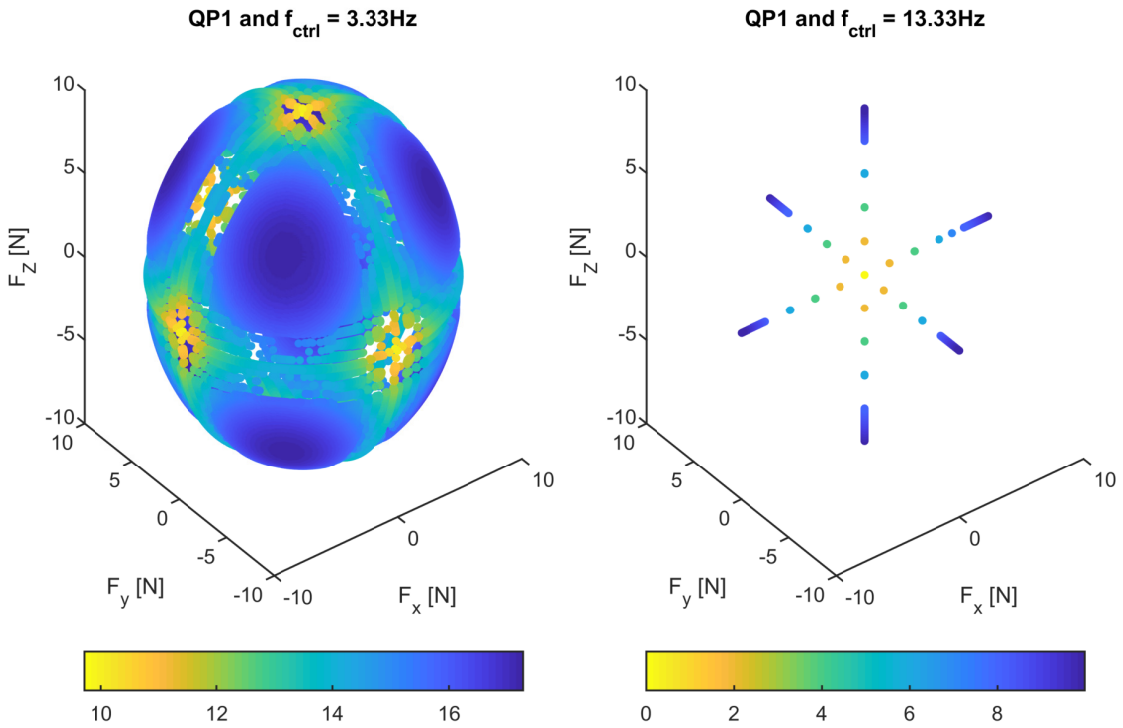


Figure 2.19: Scenario 3: Fuel consumption of actuated forces for two different control frequencies with QP1 and $f_{pwm} = 33.3kHz$

2.5.6 Fourth Scenario: Pure Torque Sampling

In this scenario, we are not sampling pure forces, but clean torques with a vector length of 10Nm. The general setup of scenario three is kept, so that we still have approximately even distributed torques on a sphere with radius 10Nm with the same amount of samples.

The same results which were previously described for clean forces can also be observed for pure torques. Each clean torque around one of the principal axes can be generated by only two thrusters. The preferred torque directions change though (see Figure 2.20). We recall that each thruster equally generates torques around two principal axes at once. Thus, it comes to no surprise that these torques can also be generated more easily. Once again, an increase of f_{ctrl} heavily skews the fuel consumption of torques on the sphere, as small deviations of optimal directions suddenly are hard to actuate. Another important fact is that due to the short lever arm, torques of the same vector length require more fuel compared to the forces.

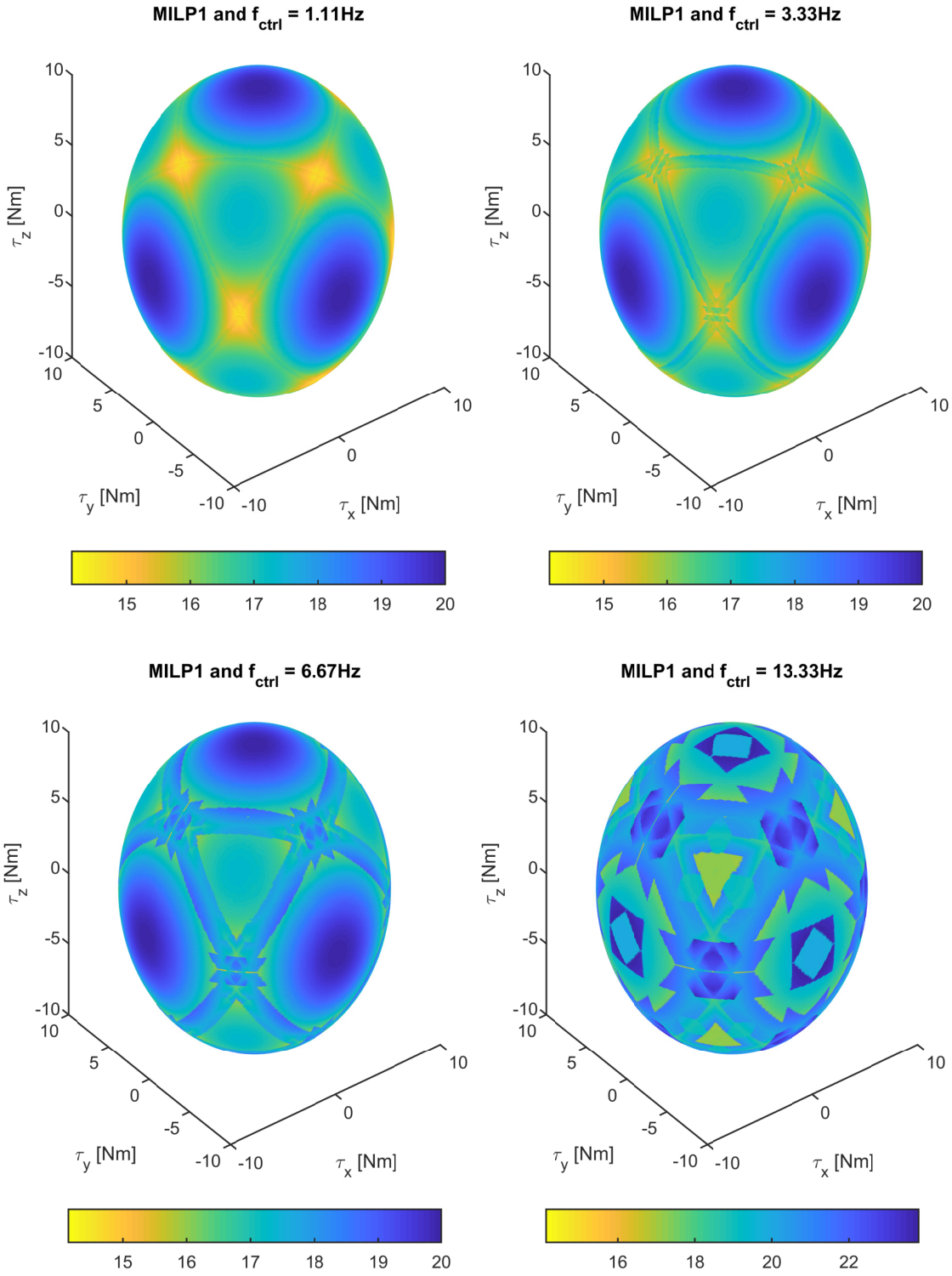


Figure 2.20: Scenario 4: Fuel consumption for different f_{ctrl} for ideal actuation of the torques on the sphere for MILP1 and $f_{pwm} = 33.3kHz$

With sub-optimal optimization methods, which do not have a notion of u_{dead} (see Figure 2.21), we observe actuation holes similarly to the ones before. Due to cutting of assignments within the dead-zone, certain torques are not actuated. These are mapped to feasible wrenches. Note that this also results in a significant actuated force, which is not represented in these plots.

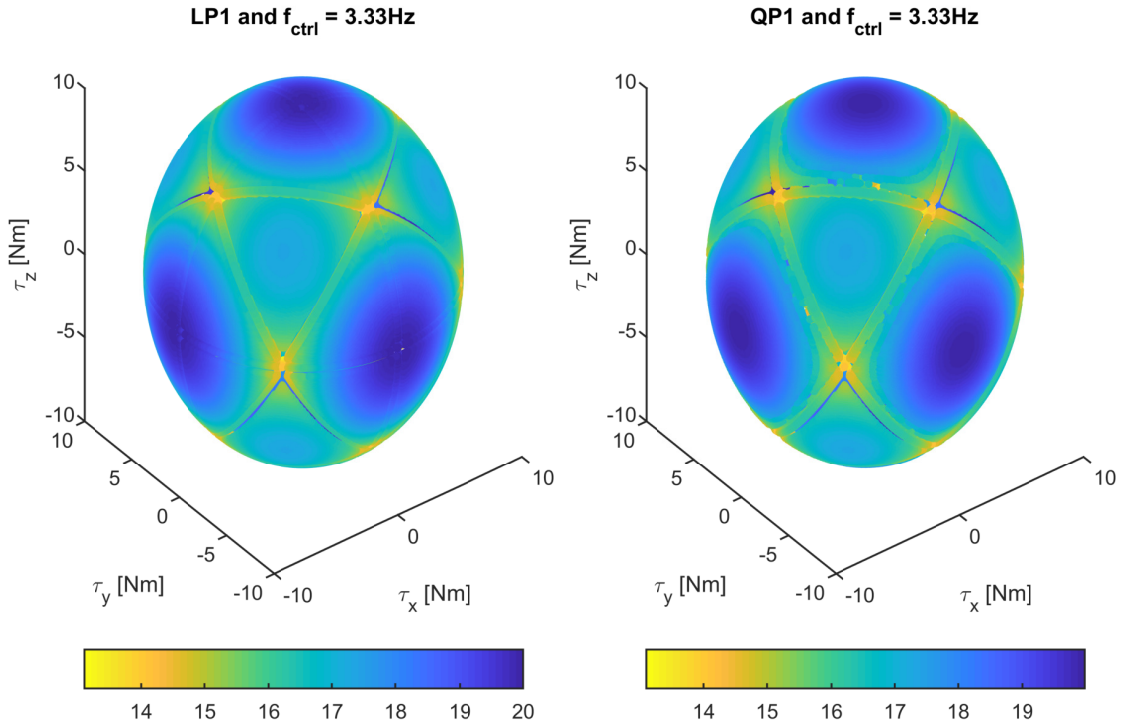


Figure 2.21: Scenario 4: Fuel consumption of actuated forces for LP1 (left) and QP1(right) and $f_{pwm} = 33.3kHz$

2.5.7 Quick-Ranking of Assignment Methods and Conclusion

We have seen that the different assignment methods have differing strengths and weaknesses. These are summarized shortly in Table 2.13. Note that not all of the previous methods are included in the list. LP2 is in all areas on par or superior to LP1. The two QP methods have a comparative performance with only a slight deviation due to the different k-value. For the used configuration, PI1 uses a worse null space vector than PI2. Thus, these three methods are not included but rank, in general, similarly to their previously listed relative method.

We use only a very rough classification. For each parameter the performance is indicated by a scale of five ranks. From worst to best, we have $[--, -, o, +, ++]$.

	LP2	LP3	MILP1	MILP2	MILP3	QP	PI2	IPI
General Wrench error	+	++	++	+	o	+	−−	o
Small wrench error	-	++	++	o	o	-	−−	-
Pure forces/torques	+	++	++	+	o	+	−−	o
Wrenches outside envelope	−−	−−	−−	−−	−−	++	+	+
Wrench overactuation in entries	o	+	+	+	++	o	−−	o
Number of thrusters	+	o	++	++	++	o	−−	o
Fuel efficiency	++	+	++	++	++	++	−−	+
Computational time	−	−	−−	−−	−−	o	++	+
Robustness of algorithm	-	-	−−	−−	−−	+	++	++

Table 2.13: Quick-ranking of assignment methods

The thruster actuation has to be selected depending on the exact mission requirements. If computational resources are scarce, it might make sense to accept larger wrench deviations and use an IPI-like approach. The static PI approaches alone proved to be very unreliable for some control wrenches, which results in large deviations. With only a few iterations using the IPI method, the quality of the assignments can be heavily improved for each control wrench and at times come even close to the LP methods for the used configuration. However, there still is a difference in the error. This enforces a higher demand on the robustness of the controller to ensure stability and accuracy.

For medium requirements on the computational speed it might be advisable to use an adapted version of the presented QP method, which is extended by a similar rounding operation as LP2. The QP methods are still significantly faster than the LP methods and are also capable to be used if the control wrenches extend the thruster envelope. However, we have to live with a slightly higher wrench error, mainly due to the more frequent use of low valued assignments.

For very high requirements on the control accuracy, only the MILP methods can ensure low wrench errors, especially for small wrenches. In addition we also select assignments which require the least amount of thrusters. A property which is disregarded by the other optimization methods. This comes at the cost of a high computational time, which also largely deviates for different control wrenches. A more in-depth investigation of the used algorithms and an optimization of these might be required before a MILP method can be reliably used to obtain the assignments. Here, starting points are a better understanding of the influence of c_{num} on the convergence speed. But also the general settings of the solver might significantly improve the run-time. A more computation efficient approach, which is also highly accurate for sufficiently high f_{pwm}, is given by LP3. On the down-side, these assignments are not optimal with respect to fuel consumption, compared to the result from MILP1.

Note that this ranking is only valid for the presented configuration and when using a PWM. For a different configuration, the performance of the different assignment methods might also change. However, even for a less redundant and non-symmetric configuration the general presented strengths and weaknesses apply. The differences might be even more developed.

Methods, which explicitly (LP3) or implicitly (MILP methods) use counter-thrust to realize shorter relative pulse firings, are more difficult to use with a PWPF or a comparable modulator. The decoupled firing intervals require a special control tuning, especially for these kinds of assignment methods.

3

Optimization of Thruster Configuration

Until now, we considered the thruster configuration to be given. However, as we saw in the previous chapter, certain parameters like the fuel consumption or the controllability envelope are shaped due to two aspects.

First, the number, position and orientation of the bounded thrusters span a physically given controllability envelope. Second, depending on control, assignment and modulation method, this physical envelope is shaped additionally. We cannot completely compensate undesired characteristics of the actuation even with the most optimal assignment and modulation methods, as the upper bound of the performance is given by the physical configuration. Thus, it is important to get an in-depth insight on how its configuration affects the controllability of a system.

The main goal of this chapter is to develop a framework that can be used to generate potential configurations and test them without the need of performing a closed-loop analysis.

By ignoring characteristics related to the assignment and modulation, the discrete nature and the dead-zone becomes irrelevant. The thrusters are simplified to positively bounded, linear actuators grouped in a configuration. As previously shown, this is valid if we assume a perfect assignment and actuation with an infinite f_{pwm}. For a sufficiently low f_{ctrl} even sub-optimal assignment methods can be used, as t_{mib} becomes irrelevant.

We will introduce some quality measures which can be used in order to optimize the thruster configuration with respect to the project requirements and geometrical pre-requisites. For each thruster we have five open parameters. The location of a thruster is identified by its translation with respect to the origin of the geometrical frame, while for the orientation it is sufficient to only give the elevation and azimuth angles due to rotational symmetry of the thrusters. We can impose several geometrical limitations which reduce the number of possible combinations of thruster poses, but in general we can locate them relatively freely on the spacecraft.

3.1 General Problem Definition

The problem of matching a control wrench with a specific thruster assignment was introduced in Subsection 2.2.1. Equation 2.3 related both properties by use of the mapping matrix **B**. Alternatively, we can express this problem as the sum of mapped thrusts for n thrusters

$$\mathbf{T}^G = \mathbf{B} \cdot \mathbf{u} = \mathbf{b}_1 \cdot u_1 + \mathbf{b}_2 \cdot u_2 + ... + \mathbf{b}_n \cdot u_n \tag{3.1}$$

We recall that assignments are only allowed to be positive. From an algebraic point of view, this means that one can identify the different vectors $\mathbf{b}_i$, which together span a positive set in the form of the controllability envelope. For now, we will ignore the fact that they are also bounded by u_{max}.

These kind of sets have already been studied since the 50s and 60s (e.g. [42]) but recently obtained new importance once again for several research directions like derivative-free optimization. We aim to use some basic properties of positive bases and positive spanning sets in order to construct our thruster configuration most optimally and preliminary check for suitability. A detailed introduction can be found in [43]. The following small introduction to positive spanning sets is based on [44].

The positive span $pos(\mathbf{S})$ of a set of vectors $\mathbf{S} = \{\mathbf{b}_1, ..., \mathbf{b}_n\} \subset \mathbb{R}^k$ is given by

$$pos(\mathbf{S}) = \{\mathbf{b}_1 \cdot u_1 + ... + \mathbf{b}_n \cdot u_n \ : \ u_i \geq 0 \text{ for all } i = 1, ..., n\} \tag{3.2}$$

Thus, formally the physical controllability envelope is the set of all positive combinations from equation 3.1 and is equal to the positive span limited by u_{max}. In the literature, $pos(\mathbf{S})$ is also often referred to as the convex cone for a subset of the k-dimensional space $\mathbb{R}^k$. On its own, each summand realizes a ray in $\mathbb{R}^k$. In order to be able to fully actuate control wrenches in all directions, we demand that the complete wrench space $\mathbb{R}^6$ is positively spanned by **S**. This equals to $k = 6$. It should be possible to generate wrenches in any direction for the unbounded thrusters. We can quickly dismiss any configuration which does not positively span $\mathbb{R}^6$. At this point we do not need any further knowledge of the boundaries of the controllability envelope, which are introduced due to the limitation by u_{max}.

Generally the k-valued vectors in **S** are called positively independent if none of them can be constructed by a positive linear combination of the others. This means that **S** is positively independent if

$$\boldsymbol{b_i} \notin pos(\mathbf{S} \backslash \{\boldsymbol{b_i}\}) \ \forall i = 1, ..., n \tag{3.3}$$

The minimal number of vectors n inside a finite set **S** that positively spans $\mathbb{R}^k$ is lower bounded by

$$|S| = n_{min} = k + 1 \tag{3.4}$$

In our thruster application this means that we require at least seven thrusters, as we have $k = 6$. The selected minimal number of vectors can be seen as one of multiple possible positive basis. A positive spanning set for $\mathbb{R}^k$ can be directly constructed from a general basis of that vector space e.g. the Euclidean basis. Then, only one additional vector is required to complete the positive basis. However, in practice this approach is not easily applicable, as the vectors $\{\mathbf{b}_1, ..., \mathbf{b}_n\}$ are reliant on the position and orientation. This means the resulting wrench vectors within the positive base cannot be chosen freely but are constrained and actually follow from the defined mounting.

As noted before, at this point we are only interested in proving if a thruster configuration positively spans $\mathbb{R}^6$. If this is not the case, we can immediately discard the generated configuration as unsuited. [44] provides us with a very basic algorithm, which allows us to check this property for a set of thrusters $\mathbf{S} = \{\mathbf{b}_1, ..., \mathbf{b}_n\} \subset \mathbb{R}^6$ with $n \geq 6 + 1$ thrusters.

1. $\mathbf{S}$ has to linearly span $\mathbb{R}^6$, which means that the mapping matrix $\mathbf{B}$ consisting of the column-wise stacked vectors in $\mathbf{S}$ requires a full row rank of 6. If the linear span condition is not fulfilled, it is impossible that $\mathbf{S}$ can positively span $\mathbb{R}^6$.
2. For each thruster $i = 1, ..., k$ we have to verify that the following equations can be fulfilled

$$-\boldsymbol{b_i} = \sum_{j=1, j\neq i}^{k} \mathbf{b}_j u_j \text{ for } u_i \geq 0 \text{ and } \forall j = 1, ...k, j \neq i \tag{3.5}$$

This equation is only given as a reference. Later on we will replace this check by an improved one using the analogy of grasping.

3.2 Sources of Inspiration

In this section, literature is presented, where approaches, how to derive an optimal thruster configuration, can be drawn from. Analogies from related fields of study are helpful as sources of inspiration and for development of the quality measures.

3.2.1 Analogy of Grasp Planning

The theory of positive linear spaces has also a role in robotic grasping problems [45]. Thus, it might be worthwhile to investigate certain solutions which are applied in grasping and try to transfer them to the thruster problem.

For the condition of the friction-less grasping of an object we have a similar problem to finding a suitable thruster configuration. The fingers in the grasping process are equal to the thrusters in the spacecraft. Without friction an object can only be manipulated by the normal forces which are applied on the boundaries of the object by the fingers.

This is equal to an applied thrust at a mounting position and therefore we can transfer the thruster mounting problem to a grasping problem. The boundary of the analogous object which has to be grasped, has to embed the constraints on the thruster positioning.

In the following, we will adapt a quantitative test of form-closure from [46] to the thruster configuration problem. For a thruster configuration, form-closure means that we can theoretically actuate all wrenches in the case of unbounded thrusters. The resulting set of equations will improve and replace our previously developed test for positive spanning thruster configurations given in equation 3.5.

The previous property of positively spanning sets can be transferred to grasping. In order to keep up our analogy to the thruster setting, we restrict ourselves to form-closure grasps. Form-closure is the ability to prevent motions of a grasped object while solely relying on uni-lateral contact constraints without friction [47]. We can imagine that a object is grasped by a hand with locked joint angles and a fixed palm in space. If the object cannot be moved, then the grasp has form-closure [48]. This property has more tightened conditions than the also often used term of force-closure. Force-closure is a relaxed property in the sense that friction forces can now contribute to the wrench.

Similar to before, we have to verify the full row rank of the mapping matrix $\mathbf{B}$ first. Also, the number of columns and thus thrusters has to be higher than the number of rows. This equals to equation 3.4, which is also referred to as Caratheodory's theorem.

We will have a non-trivial nullspace with nullspace assignments $\mathbf{u}_{null}$. Form-closure can be quantified with the following linear program formulation

$$\begin{aligned}
\max_{\mathbf{u}_{null}} \quad & d && \text{(3.6a)}\\
\text{subject to} \quad & \mathbf{B}\mathbf{u}_{null} = \mathbf{0} && \text{(3.6b)}\\
& \mathbf{u}_{null} - \mathbf{d} \geq \mathbf{0}, && \text{(3.6c)}\\
& d \geq 0, && \text{(3.6d)}\\
& \mathbf{A}\mathbf{u}_{null} \geq \mathbf{h} && \text{(3.6e)}
\end{aligned}$$

The scalar slack variable d is maximized, which is also the the measure of the form-closure. $\mathbf{d} \in \mathbb{R}^n$ is the stacked vector of values which are all equal to d. We select $\mathbf{A} = [-\mathbf{I} \;\; \mathbf{I}]^T$ with the identity matrix $\boldsymbol{I} \in \mathbb{R}^{nxn}$ and n being the number of thrusters. Accordingly we choose $\mathbf{h} = [-\mathbf{1} \;\; -\mathbf{1}]^T$. These values form an inequality, which guarantees that the solution is bounded. Thus, the maximal value of d can only be one. A value of zero means that no form-closure can be achieved. For values higher than zero we have form-closure.

Another test for form-closure can be found in [49]. In [50], it was proven that in order to guarantee form-closure, the origin of the wrench space has to be located inside the convex hull defined by the columns of the grasping matrix, which is the analogous version of $\mathbf{B}$ for the thrusters.

Equation 3.4 gave us a lower boundary for the minimally required amount of thrusters in order to be able to achieve form-closure. The equation included no limitations on the positioning of the thrusters. For a grasping problem, this equals to unconstrained arbitrary grasping of an object. However, due to the object shape and surface, the grasping is limited. Analogously, the thruster placement is also conditioned in a similar way. The theorem of Steinitz, as given in [45], specifies a maximum number of thrusters, which are required in the worst-case in order to positively span $\mathbb{R}^k$

$$n \leq 2k \tag{3.7}$$

with $k = 6$ in our case. Note that [45] uses the notion of force-closure, but the theorem can also be applied to form-closure and thus is still valid for our thruster case.

We derived a quantitative test in order to identify form-closure, which can be also used for the mapping matrix $\mathbf{B}$ and thrusters. Due to the limited scope of this work, we will restrain ourselves to this shallow introduction and one exemplary transfer of grasping solutions.

However, the grasping problem focuses not only on the evaluation of finger placement but also on optimizing the finger position and applied forces of the fingers (see e.g. [51], [52], [53] or [54]). With respect to potentially relevant performance indexes the reader is referred to the relatively recent survey [55]. All these properties are also required in the optimization of the thruster configuration. The problem in finding suitable analogies is that modern approaches often investigate the more complex task of grasping with friction. Also the number of fingers is significantly lower than the number of used thrusters. Thus, extra care has to be taken when trying to adapt grasping solutions to the space robot environment with thrusters.

3.2.2 Analogy of Swimming Robots

Another source of inspiration, which is very much related to the thruster configurations in space robots, can be found in underwater robots and vehicles. These also use thrusters as actuators and are faced with the problem of ensuring optimal manipulability with as little thrusters as possible. Typically though, these systems use bidirectional thrusters. However, we can model one bi-directional thruster by two one-directional ones, which face in opposite directions. This means that one can still draw, to some degree, conclusions from results in this area of research. We are most interested in obtaining good candidate configurations and in deriving performance measures for them.

Also, [56] investigated different designs and as a result obtained a Pythagorean solution, which ranks the best for three different quality indexes. In [57] a new one-norm algorithm was proposed in order to balance the thruster utilization. The master thesis [58] also engaged heavily in the optimization of configurations for swimming robots and can be used as reference.

3.2.3 Studies and Existing Reaction Thruster Configurations

Obviously, the most important sources are existing thruster configurations for space robots of previous projects and relating user studies, which compare different ones.

Already in the late 60s, Crawford [59] proposed insights for configuration designs. He used the linear programming solution as ground-truth and most importantly introduced the notion of level of redundancy. Also a number of different design heuristics were introduced by the author. Similarly, [60] focused on how an optimal thruster configuration has to look like, when not only mounting errors and thruster failures are present, but also the number of valves is lower then the number of thrusters. This means that not all thrusters can be actuated at the same time.

It can be helpful to simplify the problem and reduce the dimensionality. [61] investigated local controllability properties for planar bodies and thruster configurations with less than the minimally required number of four thrusters. Wang [62] proposed the notion of feasibility in control capability analysis, which allows to check if an artificial configuration fulfills the mission requirements.

For a constant flow system, [63] compared different configurations with the goal to maximize the efficiency of the used propellant. The design parameter of the envelope of least authority was developed. This means that the smallest force magnitude is compared against the moment magnitude. Jin [64] defined a margin of safety, which can be calculated for a thruster configuration and a known disturbance environment. Disturbances are seen from the perspective of maximally required forces and torques.

Relatively recently, [65] conducted a thorough study which investigated a high number of thruster configurations from attitude control systems.

3.3 Quality Measures

In the previously presented literature, as partial results, quality indexes which allow to compare different configurations more easily were developed. Different authors proposed similar indexes, so these will be condensed in the following section in order to construct our own set of quality measures. Problematic is that some indexes are not applicable in general but are dependent on specific controllers and actuation.

In this work, in order to measure the performance of a configuration, we only want to use the mapping matrix $\mathbf{B}$, which is generated from the five open parameters of each thruster. We recall that we dropped the discrete nature and the minimal firing time of the thrusters for our analysis of the configuration. Only the bounded thrust, given by u_{max}, is important in the following. Therefore, we can observe the thrusters as positively bounded, linear actuators.

Section 2.5 showed how the assignment method changes the behavior when coupled with a PWM. We saw that LP1 produces perfect assignments under the assumption of having no dead-zone, while with u_{dead} MILP1 overall gives the best assignments.

Consequently later on, we are allowed to use the assignments from LP1 as base-line in order to identify the underlying physically given minimal fuel consumption coming from the configuration. With the assumptions from above paired with no modulation, the initial LP1 assignment has the best obtainable efficiency, while in parallel results in no wrench error. MILP1 would be better overall, as it includes the effect of t_{mib}; but this makes the results dependent on f_{ctlr} and takes longer to compute.

3.3.1 Form-Closure Index

The previously developed algorithm 3.6, which determines the level of form-closure for grasping matrices, can be used as a preliminary condition for $\mathbf{B}$. Without achieving form-closure, a thruster configuration can be immediately disregarded. In addition, we also demand that configurations should have the maximal possible value for d in order to be a candidate. The value of d is stored in the index I_c.

3.3.2 Manipulability Index

An obvious choice for a suitable index is the definition of manipulability, as it can be found in general robotics research. There, the Jacobian matrix can be used in order to identify the closeness to singular points. For our mapping matrix, we will define the manipulability I_m similar to [56] as the condition number

$$I_m = cond(\mathbf{B}) = \frac{\min(\sigma_B)}{\max(\sigma_B)} \tag{3.8}$$

with σ_B being the set of singular values of the mapping matrix $\mathbf{B}$. We can see that this index is a measure of how well the thrusters are distributed. For a desired isotropic configuration, a ratio of one has to be obtained. For deviating I_m, the bounded thrusters will result in a skewed controllability envelope. For example, a very low manipulability value indicates that some wrench directions may be very difficult to realize.

One weakness of this index is that we do not use the information of all singular values. The simplest way to obtain the singular values of $\mathbf{B}$ is by calculating the SVD, which was already described in equation 2.7. If the left singular vectors in $\mathbf{U}$ coincide with the Cartesian axes, we can identify the singular values for the different force directions and analogously also the torque entries, and are capable to introduce two sub-indexes; the force and torque manipulability index

$$\begin{aligned} I_{m,f} &= \sigma_{fx} \cdot \sigma_{fy} \cdot \sigma_{fz} \\ I_{m,\tau} &= \sigma_{\tau x} \cdot \sigma_{\tau y} \cdot \sigma_{\tau z} \end{aligned} \tag{3.9}$$

We recall that a simple way to check for symmetry is to fire all thrusters at the same time and verify that this assignment is inside the nullspace. If a configuration has form-closure but $d < 1$, the columns in $\mathbf{U}$ from the SVD help to identify weak dimensions, where a reconfiguration should be done.

In [57], two methods are discussed, which can be applied in order to balance the singular values of an existing thruster configuration. One can use the results from the initial SVD to obtain $\mathbf{U}$ and $\mathbf{V}^T$. A desired distribution of singular values can be used to generate $\mathbf{S}$, which results in an adapted mapping matrix $\mathbf{B}$. Another method is described in depth in the paper and uses a one-norm algorithm. In this work though, we will not apply such a second optimization step but only use the presented indexes.

3.3.3 Control Capability

The previous two indexes give an idea, if in theory a configuration with unbounded thrusters could actuate any given control wrench (form-closure), and also, how well the different entries of the wrench vector are actuated relative to each other. However, we want to derive a parameter which explicitly expresses the Control Capability (CC) of the configuration. Previously, we also used the notion of controllability envelope for this.

A quick overview of the boundaries of the envelope is given by the maximally and minimally possible force/torque values in each direction.

$$\begin{aligned} \mathbf{I}_{b,Tmax} &= [max_{fx}(\mathbf{Bu}), max_{fy}(\mathbf{Bu}), ..., max_{\tau z}(\mathbf{Bu})] \\ \mathbf{I}_{b,Tmin} &= [min_{fx}(\mathbf{Bu}), min_{fy}(\mathbf{Bu}), ..., min_{\tau z}(\mathbf{Bu})] \end{aligned} \tag{3.10}$$

As already stated, the controllability envelope of the configuration is more complex and is only bounded by the hyper cuboid given by the previous equation.

A better property is the length and direction of the weakest wrench vector, which is defined by being on the boundary of the envelope closest to the origin. From now on this notion is abbreviated by Minimal Control Capability (minCC). One can construct a six-dimensional wrench sphere with the radius being specified by the minCC. Within this sphere all wrenches are guaranteed to be actuable.

In practice, this parameter is difficult to obtain in a closed form though. This is why there exist several approximating algorithms. We will shortly introduce the one proposed in [62], but it has to be noted that the concept of this index is also derived differently in several other literature. The notions of least authority [63] or the margin of safety [64] generally express a very similar index.

In grasp planning, there exist numerous related indexes which calculate the largest minimum resisted wrench. Though, despite extended research, no index from grasp planning that uses the exact same constraints as required in the thruster setting could be found. Other assumptions, than individually bounded forces of the fingers, are used.

With the assumption of the thrusters being bounded, linear actuators without a dead-zone behavior, the LP1 assignments perfectly express the boundaries due to the configuration. Thus, the minCC can be obtained perfectly if we can identify the wrench with the lowest radius, which becomes infeasible in the LP1 method.

The authors in [62] based their notion of minCC on an algorithm [66] which only gives a simplified assignment compared to the LP1 method. The goal of this algorithm was to obtain assignments of similar quality than the ones of LP methods but with a lower computational load. An optimal thruster combination table (OTCT) is generated offline and can be used to derive assignments online.

In short, the algorithm finds a set of different optimal combinations of six thrusters, which can be seen as optimal bases in LP theory (see similar approach in [20]). From these, one can generate sub-configurations $\mathbf{B}_k \in \mathbb{R}^{6x6}$ from the complete mapping matrix $\mathbf{B}$. If six thrusters are sufficient to actuate a desired wrench, we get the same result as with LP1. However, we get a deviation if more would be required. Therefore, this method will only obtain a reduced controllability envelope.

In short, we want to relate our previous concepts with the ones developed in the paper. The authors notion of feasibility under non-constraints (FNC) is equal to our index for form-closure. The thrusters are assumed to be unbounded, linear actuators. Analogously, feasibility under constraints (FC) includes the boundedness of the thrusters, which we denoted as the physically given controllability envelope. The assigned thrusts are only allowed to be within the range $0 < u_i < u_{max}$. Note that here the dead-zone is also not included.

The set of sub-configurations $\mathbf{B}_k$ form the OTCT and can also be exploited to derive a closed approximation of the real minCC [62]. The different $\mathbf{B}_k$ are invertible and for each optimal thruster combination the closest vector to the origin can be calculated. The overall lowest vector is equal to the approximated version of the minCC.

However, the limitation to only six active thrusters massively reduces the controllability envelope. Especially for highly redundant configurations it thus also skews the result of the minCC index. Due to this reason, we will fall back to a basic sampling process which will be explained in the following.

We will sample an arbitrary number of basic wrench vectors $\mathbf{y}_i$, which are located on the six-dimensional unit sphere and are grouped into a matrix $\mathbf{Y}$. The sampling process can be efficiently carried out with the help of a 6-dimensional Gaussian distribution with the identity matrix as covariance matrix. Effectively, we draw six independent, normally distributed random variables $\mathbf{x}_i = [x_{i1}, x_{i2}, x_{i3}, x_{i4}, x_{i5}, x_{i6}]$ for each sample. A basic wrench vector is obtained by scaling each sample with the Euclidean norm

$$T_{ik} = \frac{x_{ik}}{||\mathbf{x}_i||_2} \tag{3.11}$$

with $k = 1...6$ and $\mathbf{T}_{s,i} = [T_{i1}, T_{i2}, T_{i3}, T_{i4}, T_{i5}, T_{i6}]$. This sampling process uses the fact that the multivariate normal distribution with an identity as covariance matrix is rotationally symmetric around the origin.

We can formulate a new type of linear optimization problem in order to obtain the boundary of the wrench for the direction defined by the sampled basic wrench.

$$\begin{aligned} &\max && d_i && (3.12a)\\ &\text{subject to} && \mathbf{B} \cdot \mathbf{u} = d_i \cdot \mathbf{T}_{s,i}, && (3.12b)\\ &&& 0 \leq u_k \leq u_{max} \quad \text{for } k \in [1...n] && (3.12c) \end{aligned}$$

B still represents the mapping matrix defined by the configuration and **u** is the resulting assignment for all of the n thrusters. We are not interested in the most fuel efficient assignment, but only demand that one feasible assignment exists. This optimization problem has to be solved for each sample. For an infinite amount of samples and thus directions, we would obtain the exact boundary of the controllability envelope.

Due to the generation process of the samples, the parameter d_i also represents the distance from the origin. This assumes that the meaning of the values of forces and torques is equal. However, one could simply adopt a different distance measure by directly using the obtained boundary wrenches. Depending on the thruster configuration and design goals, a different weighting of each wrench dimension might be sensible as well. In theory, the same result would be obtained when we directly sample from an six-dimensional ellipse instead of a sphere.

For the stacked distance values **d** of the N random sample directions, we can derive the following indexes

$$\begin{aligned} I_{d,min} &= \min(\mathbf{d}) \\ I_{d,max} &= \max(\mathbf{d}) \\ I_{d,avg} &= \frac{1}{N} \sum_{i=1}^{N} d_i \\ I_{d,std} &= \sqrt{\frac{1}{N-1} \sum_{i=1}^{N} |d_i - I_{d,avg}|^2} \end{aligned} \tag{3.13}$$

Additionally, we can identify the wrenches where the minimal and maximal distance was obtained, with

$$\begin{aligned} \boldsymbol{I}_{T,min} &= d_{min} \cdot \mathbf{T}_{s,min} \\ \boldsymbol{I}_{T,max} &= d_{max} \cdot \mathbf{T}_{s,max} \end{aligned} \tag{3.14}$$

These minimal and maximal distances are not necessarily unique.

As a partial result of the generated boundary wrenches, we can approximately calculate the volume I_v of the controllability envelope. Note that once again this value has to be taken with a grain of salt, as this index mixes torques and forces. The easiest way to achieve this is by calculating the convex hull of all samples and use the result for the volume calculation.

It has to be noted that we could obtain $I_{d,max}$ and $I_{T,max}$ during the following optimization process using the stacked parameter $\mathbf{x} = [\mathbf{u} \;\; \mathbf{T}]$

$$\begin{aligned} \max \quad & \mathbf{x}^T \mathbf{H} \mathbf{x}^T && \text{(3.15a)} \\ \text{subject to} \quad & [\mathbf{B} \;\; -\mathbf{1}] \cdot \mathbf{x} = \mathbf{0}, && \text{(3.15b)} \\ & 0 \leq u_k \leq u_{max} \quad \text{for } k \in [1...n] && \text{(3.15c)} \\ & -\infty \leq T_j \leq \infty \quad \text{for } j \in [1...6] && \text{(3.15d)} \end{aligned}$$

with $\mathbf{1} \in \mathbb{R}^{6x1}$ and $\mathbf{H} = [\mathbf{0} \;\; \mathbf{I}_{6x6}]$. We can see that both the maximum wrench and the corresponding assignment are unknown and therefore used as an optimization variable here. Note that this problem is non-convex and thus requires a special solver. For example, the function *fmincon* in MATLAB can be used.

3.3.4 Distribution of Fuel Consumption

It was already introduced that we will inherently have wrench directions which can be actuated more fuel conservative than others. Though intuitively, wrenches with the same distance from the origin should require the same amount of fuel.

Similar to the approach conducted in the control capability indexes we will sample from the 6-dimensional unit sphere. This means that we are using, once again, the Euclidean norm as distance measure with a same weighting of the forces and torques. In addition, we now scale the radius to a desired value. For example, the nominal thrust of one thruster would be an obvious choice. By plugging in the sample wrenches into the LP1 method, we will obtain the solution with the lowest possible fuel consumption given the specific configuration. In order to derive a meaningful interpretation on how these consumptions are distributed on the sphere, we will again use indexes obtained with min-, max- and mean-operations.

However, as we saw before, when having both forces and torques present, the concept of distances becomes more complex, as we have to normalize the effect of the different dimensions. This is why in Subsection 2.5.5 we additionally instituted the concept of clean torques and forces. With these, we can intuitively use the L2-norm, which results in a sampling of defined forces/torques with a distinctive radius. The other part of the wrench is then set to zero. These two sets of wrenches are sub-sets of the complete wrench space and will be used additionally.

To summarize, we sample from three different surfaces. One is a six-dimensional body with a particular radius and a potentially different weighting of both forces and torques (Fuel index 1). The other two surfaces are the three-dimensional force and torque sets, where the respective other parameters are set to zero (Fuel indexes 2 and 3). For each of the sampled wrenches in each of the three scenarios we save the sets $\mathbf{L}_1$, $\mathbf{L}_2$, $\mathbf{L}_3$ of calculated fuel consumptions for each sample $L_i = \mathbf{k}^T \mathbf{u}_i$ with $\mathbf{k} = \mathbf{1}$.

For the example of scenario 1, the calculated indexes are given in the following

$$\begin{aligned} I_{f1,min} &= \min(\mathbf{L}_1) \\ I_{f1,max} &= \max(\mathbf{L}_1) \\ I_{f1,avg} &= \frac{1}{N}\sum_{i=1}^{N} L_i \\ I_{f1,std} &= \sqrt{\frac{1}{N-1}\sum_{i=1}^{N} |L_i - I_{f1,avg}|^2} \end{aligned} \tag{3.16}$$

for N samples. This is also done accordingly in the other two surfaces.

3.3.5 Level of Redundancy

We not only want to optimize our thruster configurations in terms of fuel efficiency and wrench performance, but ideally also want to be able to actuate every wrench direction even if one or more thrusters or their corresponding valves break down. Once again it has to be stressed that in this work we assume that each thruster has its own valve (compared to e.g. [60]).

Crawford [59] already introduced a notion of redundancy in his work, where he stated how many thrusters are required for different levels of redundancy. He stated that for a k-dimensional task space, we require at least two additional thrusters per level of redundancy. This results in the minimal amount of n thrusters for a specific redundancy level of r according to

$$n = k + 1 + 2 \cdot r \tag{3.17}$$

Note that this equation also includes the equation for the minimal number of thrusters to obtain form-closure.

We will use a slightly different notion of redundancy. For a specific level of redundancy I_r, the specified number of thrusters should be allowed to malfunction without losing the property of form-closure. For example, for $I_r = 2$, all possible combinations of two malfunctioning thrusters should have no effect on the form closeness.

To further differentiate configurations with the same level of redundancy, in addition we will track the ratio of failing thruster combinations where form-closure is lost compared to the ones where this is not the case. This fracture is added as a value after the decimal point and shows how close we are to the next level of redundancy.

To continue our example, all combinations of configurations with two arbitrary malfunctioning thrusters still remain form-closure, but at half of the combinations of three thrusters do not. This results in a value $I_r = 2.5$.

3.4 Two-Dimensional Thruster Configurations

Before investigating different three-dimensional configurations, we want to first see the effect of additional thrusters to a minimal configuration in the two-dimensional plane. Only forces in x- and y-direction and a torque around the z-axis are possible. This consideration allows to visualize the shape of the controllability envelope. Four different configurations will be introduced and the previously developed indexes will be calculated.

With only two forces and one torque present in the control wrench, we can see that a minimum of four thrusters is required. An example for a possible minimal configuration is given by $\boldsymbol{B}_1$. This configuration is obtained by placing the thrusters on the four corners of a square with side length of $1m$ and rotating them in different directions by $45°$. The Center of Mass (COM) is assumed to be perfectly located at the center, where also the origin of the geometrical frame is located. The positioning of the thrusters resembles the one of the 3D-case depicted in Figure 2.10 but only for two dimensions.

This basic configuration is extended in configurations $\mathbf{B}_2$ and $\mathbf{B}_3$ by four additional thrusters. In $\mathbf{B}_2$ we will locate the new thrusters in a way that the new configuration equals the one given by four bi-directional thrusters.

$$\mathbf{B}_1 = \begin{bmatrix} -f_t & -f_t & f_t & f_t \\ -f_t & f_t & f_t & -f_t \\ f_t & -f_t & f_t & -f_t \end{bmatrix} \qquad \mathbf{B}_2 = \begin{bmatrix} -f_t & -f_t & f_t & f_t & -f_t & -f_t & f_t & f_t \\ -f_t & f_t & f_t & -f_t & -f_t & f_t & f_t & -f_t \\ f_t & -f_t & f_t & -f_t & -f_t & f_t & -f_t & f_t \end{bmatrix}$$

where $f_t = \cos(45°) = \frac{\sqrt{2}}{2}$.

In comparison, $\mathbf{B}_3$ is enlarged by four thrusters which realize no torque but only clean negative and positive x- and y-forces. Thus, they have to be located without lever arm on the axes starting from the COM.

$$\mathbf{B}_3 = \begin{bmatrix} -f_t & -f_t & f_t & f_t & 1 & 0 & -1 & 0 \\ -f_t & f_t & f_t & -f_t & 0 & 1 & 0 & -1 \\ f_t & -f_t & f_t & -f_t & 0 & 0 & 0 & 0 \end{bmatrix}$$

Without redundancy and a low number of thrusters, only configurations with thrusters producing coupled wrench vectors will lead to form-closure. However it is not entirely clear, if these configurations are still better if more than the minimal amount of thrusters are used. Therefore, as a comparison we use configuration $\mathbf{B}_4$ in the intuitive two-dimensional example

$$\mathbf{B}_4 = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 & -1 & 0 & -1 \\ -1 & 0 & 0 & 1 & 1 & 0 & -1 & 0 \\ 0.5 & -0.5 & 0.5 & -0.5 & 0.5 & -0.5 & -0.5 & 0.5 \end{bmatrix}$$

In this configuration the thrusters are still located on the corners of the unit square (two at each), but rotated by multiples of $90°$. This means that we have a clear lever arm of $0.5m$.

Form-Closure and Manipulability All presented configurations fulfill the condition of form-closure. They even reach the maximum value in the quantitative form-closure test, which leads to $I_c = 1$ for all configurations. As discussed, this also means that the left singular vectors of the SVD can be simply matched to be along the positive or negative axis for each dimension (e.g. $[1\ 0\ 0]$). This is why we are able to not only compute the general manipulability index I_m, but also are able to identify the singular values which correspond to each force and torque.

In both configuration $\mathbf{B}_1$ and $\mathbf{B}_2$, all singular values are the same, which means that there will be no preferences in the wrench entries. Both forces and the torque can be actuated the same way when only considering their absolute values. Due to double the amount of thrusters being used, $\mathbf{B}_2$ has higher singular values. The numerical results of the indexes are part of Table 3.1.

In contrast $\mathbf{B}_3$ and $\mathbf{B}_4$ have different lever arms. The singular value matched with the torque around the z-axis is lower than the other two corresponding to the forces. Lower absolute values of torque can be generated, compared to the forces.

	I_c	I_m	$I_{m,f}$	$I_{m,\tau}$	$I_{d,min}$	$I_{d,max}$	$I_{d,avg}$	$I_{d,std}$	I_v
$\mathbf{B}_1$	1	1	2.83	1.41	10	14.13	11.01	0.72	5.73e3
$\mathbf{B}_2$	1	1	4	2	20	28.25	22.01	1.43	4.53e4
$\mathbf{B}_3$	1	0.71	4	1.41	14.14	26.12	19.46	2.93	3.31e4
$\mathbf{B}_4$	1	0.71	4	1.41	16.33	28.28	19.43	2.28	3.22e4

Table 3.1: Part 1: Index values for two-dimensional configurations

Control Capability It is obvious that the torque control capability is directly depending on the lever arm and will grow with its length. Therefore, it makes sense to locate the thrusters, in order to be able to generate larger torque values, in a way so that they have the largest possible distance from the origin. It is assumed that the used geometry for the thruster location is equal to the maximal possible distance to the frame origin.

Figure 3.1 visualizes the calculated convex hulls of the set of sampled test directions. The resulting body is described by a polygon.

Both $\mathbf{B}_1$ and $\mathbf{B}_2$ are symmetric to all axes. $\mathbf{B}_2$ effectively doubles the size of the envelope, which can be also seen in the values of their indexes (see Table 3.1). The weakest wrench direction, which is the closest to the origin, can be identified by $[-7.07N\ 0N\ 7.07Nm]$. For $\mathbf{B}_2$ this vector is scaled by the factor 2.

The additional thrusters in $\mathbf{B}_3$ compared to $\mathbf{B}_1$ only add additional clean forces. This is why the torque cannot be enlarged in value and the weakest directions are clean torques. However, we can significantly enlarge the amount of feasible forces.

The decoupled configuration $\mathbf{B}_4$ can only generate smaller torques than $\mathbf{B}_2$. Thus, its weakest directions are given by a combination of all parts of the wrench (e.g. $[6.67N\ 6.67N\ -13.33Nm]$). However $\mathbf{B}_4$ can produce larger wrenches when only forces are required. In addition, it is less skewed than B_3, which means the standard deviation of the distance is lower.

Overall, when using I_v as reference, we can see that, in volume, $\mathbf{B}_2$ has the largest control envelope. However, as already noted, this property ignores the difference in the wrench units.

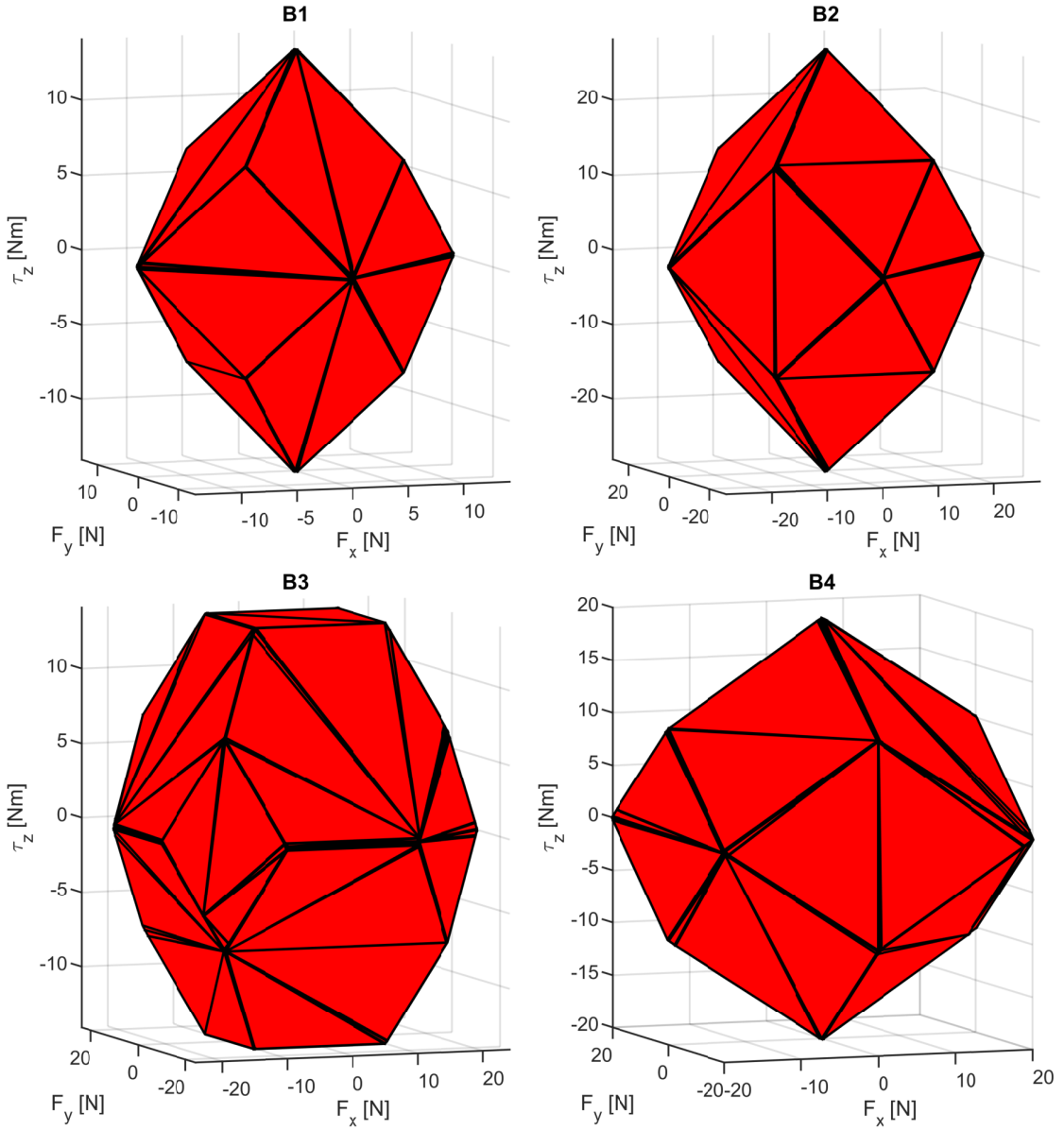


Figure 3.1: Convex hull of controllability envelope for given two-dimensional configurations

Fuel Distribution on Wrench Sphere By calculating the fuel consumption for wrenches on the sphere with a radius of the nominal thrust $u_{max} = 10N$, we can visualize and understand the concept of more fuel-optimal directions. If we look at the produced wrench of each thruster as a vector, then along this vector the most fuel conservative wrenches are found.

This is why in Figure 3.2, for example $\mathbf{B}_2$ has 8 of these, while $\mathbf{B}_1$ has only 4. We can greatly reduce the spread of the fuel consumptions on this sphere if new thrusters are added to previously bad wrench directions. $\mathbf{B}_2$ basically did this for the configuration $\mathbf{B}_1$. Therefore, the mean fuel consumption and the standard deviation can be greatly reduced (see Table 3.2). This motivates a possible configuration design. We start from a minimal configuration and add thrusters in each worst fuel direction until the result is satisfactory. For an infinite amount of iterations, all wrench directions would require the same amount of fuel.

In configuration $\mathbf{B}_4$, the thruster distribution also leads to 8 preferred directions. However, we can see that from the point of view of minimal fuel consumption this configuration is worse than $\mathbf{B}_2$. The presence of large torques is very fuel consuming in $\mathbf{B}_4$, while wrenches with a small torque require approximately the same fuel as $\mathbf{B}_2$. This is also verified by the values in Table 3.2. $\mathbf{B}_2$ performs the best overall with respect to indexes corresponding to the fuel consumption.

Note that the fuel consumption indexes, when only clean forces and torques are present, were not calculated explicitly for the 2D configuration.

	$I_{f,min}$	$I_{f,max}$	$I_{f,avg}$	$I_{f,std}$	I_r
$\mathbf{B}_1$	8.18	24.49	18.23	3.78	0.00
$\mathbf{B}_2$	8.17	14.14	11.75	1.42	1.57
$\mathbf{B}_3$	8.17	20.00	14.97	2.94	1.64
$\mathbf{B}_4$	8.94	20	13.61	2.63	1.57

Table 3.2: Part 2: Index values for two-dimensional configurations

Level of Redundancy Obviously, as $\mathbf{B}_1$ only has the minimal amount of required thrusters for the given task, it is not redundant. Consequently we have $I_r = 0.0$. The other configurations however, are capable to still actuate any wrench direction, even when one thruster breaks down. Note that a thruster failure still results in a reduced controllability envelope.

Though, in all these configurations it is not possible to guarantee form-closure if two arbitrary thrusters break down. There exist possible combinations of erroneous thrusters, which result in a loss of actuable wrench directions. This is the case if, for example, in $\mathbf{B}_2$ the two thrusters located on any of the four corners break down. For configurations of 8 thrusters there exist 28 possibilities of two simultaneous thruster failures. Overall we can identify 12 combinations for $\mathbf{B}_2$ which result in a loss of form-closure. Analogously, $\mathbf{B}_4$ has the same amount of problematic combinations, while $\mathbf{B}_3$ has only 10.

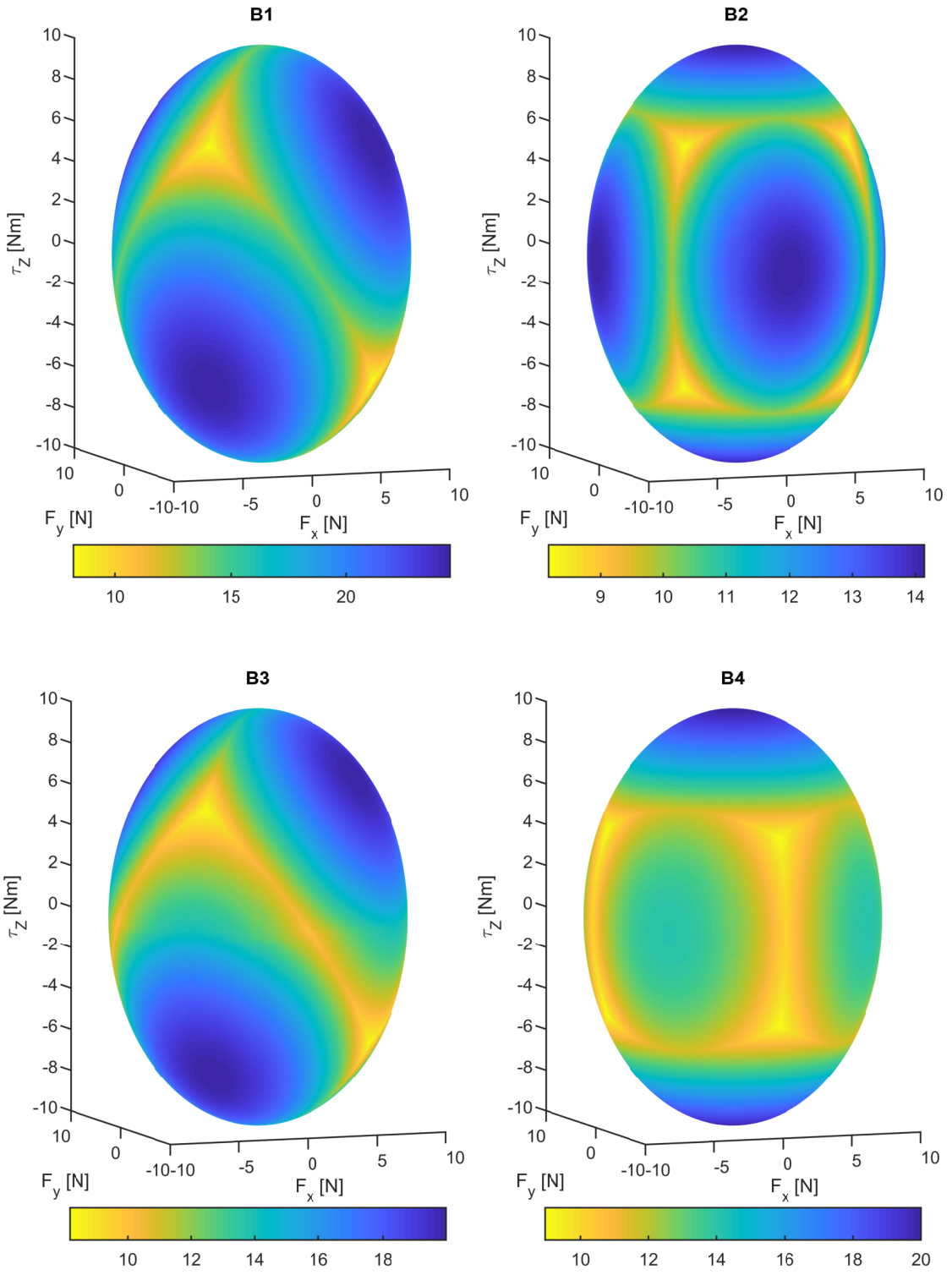


Figure 3.2: Minimal fuel consumptions for given two-dimensional configurations

3.5 Three-Dimensional Thruster Configurations

As hinted before, a three dimensional thruster configuration leads to five open parameters for each thruster. In theory, without any constraints on the thruster pose, we have an endless number of possible combinations of thruster poses. As stated, it is sensible to locate the thrusters at positions with the largest distance to the center in order to generate the highest amount of torques. We assume, similar to the two-dimensional setting, that the spacecraft body has the shape of a cube. The length of each side is $2m$ with the origin of the geometrical frame being at the center.

This still leaves us with an infeasible number of candidate configurations. Therefore, we limit the position of a thruster to only be located on one of the eight corners (similar to Figure 2.10). We want to generate configurations with different levels of redundancy. As a result, this approach will lead to configurations with either 8, 16 or 24 thrusters and accordingly with either one, two or three thrusters on each corner. The fixation of the thruster positions reduces the number of open parameters for each thruster to only two, the azimuth and elevation angle.

The space of candidate configurations can be further limited when we limit the set of orientations on each corner. Thrusters cannot fire towards the structure without potentially damaging it. Also, if the thruster firing directions intersect after a short distance, the effectively applied thrust might be reduced. We will take a very conservative approach in the following, which guarantees that the thrusts cannot intersect at all along the line of firing.

Finally, we introduce a last restriction. Configurations that are geometrically symmetric to the planes spanned by the Cartesian axes heavily simplify the analysis and in addition exhibit some advantageous properties. For example, clean actuation in one of the six entries is easily possible through combinations of multiple thrusters leading to the generation of a high number of form-closure configurations. As a side-effect, by selection of the orientation of only one thruster on any corner, all the other thruster orientations follow accordingly. Only two parameters have to be selected to generate a configuration. This will become more comprehensive after the next subsection.

3.5.1 Minimal Configurations with 8 Thrusters

At least seven thrusters are required in order to obtain a minimal configuration with form-closure.

In our approach with fixed thruster locations we will use 8 thrusters, one on each corner of the cube. This means that the position of each thruster is given by the displacement vector $\mathbf{r}$

$$\mathbf{r} = \begin{bmatrix} 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \end{bmatrix}$$

The configuration is defined to be symmetric to the three planes spanned by the Cartesian axes (e.g. the x-y-plane). Thus, we only have to define the azimuth and elevation angle of one thruster; we will use the one at position r_1 and mirror the resulting pose. We recall that our definitions of the orientation angles of the thrusters were defined so that the wrench vector is obtained directly. It is important to keep in mind that this is the opposite firing direction and, in order to guarantee no intersecting thrusts with either one other thruster or the structure, we have to limit the azimuth angles to the set $\chi_1 \in \left]-\pi, ..., -\pi + \frac{\pi}{2}\right]$ and elevation angle to $\gamma_1 \in [-\frac{\pi}{2}, ..., 0]$. Only one eighth of the orientation space is used.

Once again a sampling approach is used to generate potential configurations. Both sets are dissected with a resolution of 0.1° and combined to obtain over 800,000 candidate configurations. The calculation of all performance indexes for all candidates is computationally very expensive. Thus, we first check for form-closure and in a second step use the manipulability indexes to compare the different candidates, which have form-closure.

Figure 3.3 shows the result of the sampled orientation space when the manipulability index, defined in equation 3.8, is used. Most of the candidates exhibit form-closure. Only at the boundaries and, for example, for $\chi_1 = -135^\circ = 225^\circ$, the mapping matrix $\mathbf{B}$ looses rank and thus is not form-closed. The manipulability goes towards zero. For the given setup of configurations, we can see that it is impossible to weight each wrench entry equally. The smallest singular value will, at most, be a third of the largest one. We can identify six configurations which have the best ratio of lowest to highest singular values.

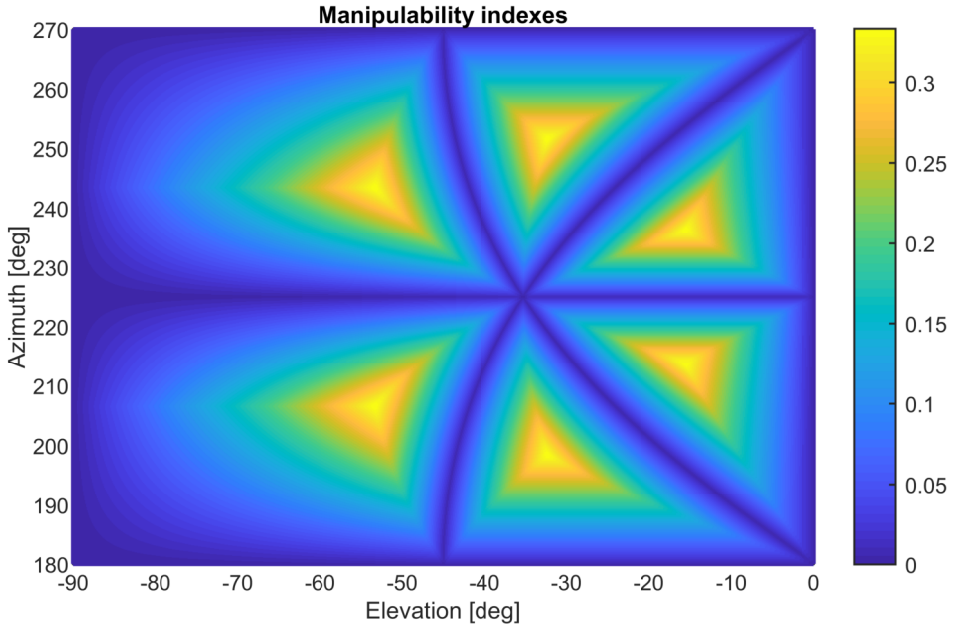


Figure 3.3: Manipulability I_m for symmetric 8 thruster configurations

The configuration below is one of these and can be obtained for $\chi_1 = -146.3°$ and $\gamma_1 = -15.5°$.

$$\mathbf{B}_{8Thr,1} = \begin{bmatrix} -0.80 & -0.80 & 0.80 & 0.80 & -0.80 & -0.80 & 0.80 & 0.80 \\ -0.53 & 0.53 & 0.53 & -0.53 & -0.53 & 0.53 & 0.53 & -0.53 \\ -0.27 & -0.27 & -0.27 & -0.27 & 0.27 & 0.27 & 0.27 & 0.27 \\ 0.27 & -0.27 & -0.27 & 0.27 & -0.27 & 0.27 & 0.27 & -0.27 \\ -0.53 & -0.53 & 0.53 & 0.53 & 0.53 & 0.53 & -0.53 & -0.53 \\ 0.27 & -0.27 & 0.27 & -0.27 & 0.27 & -0.27 & 0.27 & -0.27 \end{bmatrix}$$

Note that the other five optimal configurations will contain the same row entries, but only permuted in order. We permute the force and torque rows jointly, for example, the first row with the second and the fourth one with the fifth, to generate the other candidates.

Similarly to the two-dimensional case, fully coupled wrench vectors have to be used for a minimal configuration. It is obvious that a pure force or torque can be exerted by actuation of four thrusters at the same time.

Keeping in mind the task of space robots, the presented configuration was deemed the most suitable one out of these six. It is assumed that a translational movement of the space robot will occur primarily in x-direction, which means that it is advantageous to locate a large portion of the thrust in this direction. In addition, an interaction with a satellite through the robot arm will primarily lead to torques around the y-axis.

We can directly map the singular values to a specific force or torque if the left singular vectors are equal to signed unit vector. This allows to calculate force and torque manipulability indexes as given by the equations in 3.9. Figure 3.4 shows the result for the multiplication of $I_{m,f}$ and $I_{m,\tau}$, which means that all singular values are multiplied in the process. Still, six optimal configurations are obtained, which also have all the same row entries. However it gets apparent that the optima are slightly displaced compared to before. The optimal azimuth angles are marginally different.

For example, for the optimum $\chi_1 = -151.2°$ and $\gamma_1 = -15.5°$ we get the configuration matrix

$$\mathbf{B}_{8Thr,2} = \begin{bmatrix} -0.86 & -0.86 & 0.86 & 0.86 & -0.86 & -0.86 & 0.86 & 0.86 \\ -0.47 & 0.47 & 0.47 & -0.47 & -0.47 & 0.47 & 0.47 & -0.47 \\ -0.17 & -0.17 & -0.17 & -0.17 & 0.17 & 0.17 & 0.17 & 0.17 \\ 0.30 & -0.30 & -0.30 & 0.30 & -0.30 & 0.30 & 0.30 & -0.30 \\ -0.69 & -0.69 & 0.69 & 0.69 & 0.69 & 0.69 & -0.69 & -0.69 \\ 0.39 & -0.39 & 0.39 & -0.39 & 0.39 & -0.39 & 0.39 & -0.39 \end{bmatrix}$$

Compared to the previous configuration result, we can see that we will have more control capability for forces in x-direction and also higher maximal torques. This comes at the cost of a reduced controllability for forces along the y- and z-axis.

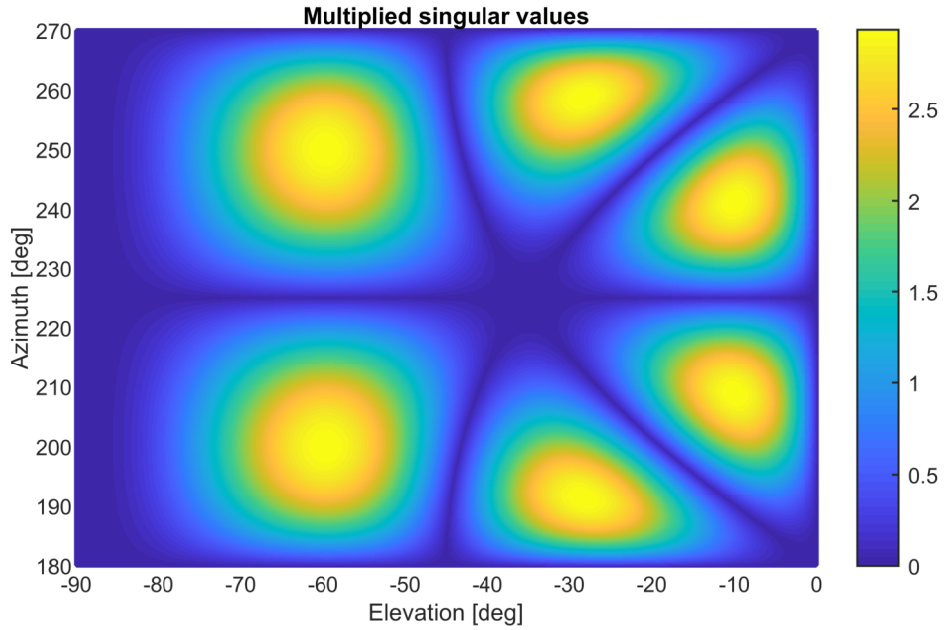


Figure 3.4: Product $I_{m,f} \cdot I_{m,\tau}$ for symmetric 8 thruster configurations

For both of the obtained candidate configurations, a more in-depth performance measurement will be done in Section 3.6, where also the other indexes are calculated.

3.5.2 Pythagorean Solution with 12 Thrusters

Even though the locations on the corners of the cube will result in the largest distance to the center, depending on the orientation the maximum torques may not be generated.

One could also locate the thrusters in the center of one of the six faces of the cube. Pierrot [56] investigated different thruster configurations for underwater vehicles with the result that this approach shows the best overall results. He derived a Pythagorean configuration, which generates more decoupled wrench vectors. It has to be noted that he used bi-directional thrusters, which is why we use two one-directional thrusters to generate the same functionality. Therefore, the resulting configuration $\mathbf{B}_{12Thr,3}$ requires overall 12 thrusters and is given by

$$\mathbf{B}_{12Thr,3}(1:6) = \begin{bmatrix} 0 & 0 & 0.53 & 0.53 & 0.85 & -0.85 \\ 0.85 & -0.85 & 0 & 0 & 0.53 & 0.53 \\ 0.53 & 0.53 & 0.85 & -0.85 & 0 & 0 \\ 0 & 0 & 0.85 & 0.85 & -0.53 & 0.53 \\ -0.53 & 0.53 & 0 & 0 & 0.85 & 0.85 \\ 0.85 & 0.85 & -0.53 & 0.53 & 0 & 0 \end{bmatrix}$$

$$\mathbf{B}_{12Thr,3}(7:12) = \begin{bmatrix} 0 & 0 & -0.53 & -0.53 & -0.85 & 0.85 \\ -0.85 & 0.85 & 0 & 0 & -0.53 & -0.53 \\ -0.53 & -0.53 & -0.85 & 0.85 & 0 & 0 \\ 0 & 0 & -0.85 & -0.85 & 0.53 & -0.53 \\ 0.53 & -0.53 & 0 & 0 & -0.85 & -0.85 \\ -0.85 & -0.85 & 0.53 & -0.53 & 0 & 0 \end{bmatrix}$$

3.5.3 Redundant Solution with 16 Thrusters

The thrusters in all of the previous configurations produce highly coupled wrenches. This is necessary in order to generate form-closed configurations even with a low number of thrusters. With an increased number of thrusters, it is possible to decouple the thrust vectors. One thruster only produces a force along one axis and also only torques around specific axes. In addition, one can increase the level of redundancy. A higher number of thrusters can also improve the controllability envelope and the average fuel consumption.

In this subsection we will increase the number of thrusters on each corner by one and thus obtain a configuration of 16 thrusters. Still, we constrain ourselves to configurations which are symmetric to the Cartesian planes. Thus, we have only four open parameters. For example, both thrusters u_1 and u_9 are located at r_1 and their orientation has to be specified. All poses of the other thrusters can be derived from this.

The same constraint set as for the 8 thrusters is used but now divided into two distinct subsets. The orientation of the first thruster is drawn from $\chi_1 \in \left]-\pi, ..., -\pi + \frac{\pi}{4}\right]$ and $\gamma_1 \in [-\frac{\pi}{2}, ..., 0]$, while thruster nine is chosen from $\chi_9 \in \left]-\pi + \frac{\pi}{4}, ..., -\pi + \frac{\pi}{2}\right]$ and $\gamma_9 \in [-\frac{\pi}{2}, ..., 0]$.

In order to reduce the number of samples, we will not draw each parameter independently from each other. The unconstrained azimuth and elevation angle span a sphere of force directions. Similar to Subsection 2.5.5, we can approximately evenly distribute the samples on the sphere by triangulation. The resulting samples for the two thrusters are combined independently, which results in over six million candidates.

Once again, not all indexes will be calculated for all of these candidates, but the notions of form-closure and manipulability are used to pre-select potentially good candidates.

This time, overall nine candidate configurations could be derived. We stack these candidates in a matrix with each row representing one configuration with $[\gamma_1\ \chi_1\ \gamma_9\ \chi_9]$ given in degree

$$\begin{bmatrix} -45 & -180 & 0 & -135 \\ -45 & -180 & -45 & -90 \\ 0 & -135 & -45 & -90 \\ -90 & -180 & 0 & -112.5 \\ 0 & -180 & -67.5 & -90 \\ -22.5 & -180 & 0 & -90 \\ 0 & -180 & -22.5 & -90 \\ -67.5 & -180 & 0 & -90 \\ 0 & -157.5 & -90 & -135 \end{bmatrix}$$

We recall that these two thrusters are defined to be located on the corner r_1.

The first three candidates optimize the manipulability index and result in $I_m = \frac{1}{\sqrt{2}}$. Once again it is impossible to get the same controllability boundaries for all wrench entries, as we always have $I_m < 1$. We recall that it is advantageous to have a stronger force for f_x and larger capabilities for τ_y. The setup from the first three solutions however prohibits that both are optimized. Thus, we just select the first configuration in the given list which has larger f_x and τ_x entries. This means we have a selected final candidate equal to

$$\mathbf{B}_{16Thr,A}(1:8) = \begin{bmatrix} -0.71 & -0.71 & 0.71 & 0.71 & -0.71 & -0.71 & 0.71 & 0.71 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -0.71 & -0.71 & -0.71 & -0.71 & 0.71 & 0.71 & 0.71 & 0.71 \\ -0.71 & 0.71 & 0.71 & -0.71 & 0.71 & -0.71 & -0.71 & 0.71 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.71 & -0.71 & 0.71 & -0.71 & 0.71 & -0.71 & 0.71 & -0.71 \end{bmatrix}$$

$$\mathbf{B}_{16Thr,A}(9:16) = \begin{bmatrix} -0.71 & -0.71 & 0.71 & 0.71 & -0.71 & -0.71 & 0.71 & 0.71 \\ -0.71 & 0.71 & 0.71 & -0.71 & -0.71 & 0.71 & 0.71 & -0.71 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.71 & -0.71 & -0.71 & 0.71 & -0.71 & 0.71 & 0.71 & -0.71 \\ -0.71 & -0.71 & 0.71 & 0.71 & 0.71 & 0.71 & -0.71 & -0.71 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Each thruster will only manipulate four entries of the wrench. Depending on the combination of overall three different groups of eight thrusters, we obtain either the first, the second or the third configuration given by the list 3.5.3.

Alternatively, we can not try to homogenize the envelope but generate the largest product of singular values from the configuration. This approach also indirectly optimizes the volume of the controllability envelope. Starting from the fourth solution in the given candidate list, we can identify six possible configurations which maximize this property. Once again, all result in the same rows in the mapping matrix $\mathbf{B}$, only with permuted indexes. We select the last solution as candidate to be further investigated. This configuration results in a large feasible torque τ_y and an acceptable f_x. It has to be noted that f_y will be relatively weak in its maximal actuation.

$$\mathbf{B}_{16Thr,5}(1:8) = \begin{bmatrix} -0.92 & -0.92 & 0.92 & 0.92 & -0.92 & -0.92 & 0.92 & 0.92 \\ -0.38 & 0.38 & 0.38 & -0.38 & -0.38 & 0.38 & 0.38 & -0.38 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.38 & -0.38 & -0.38 & 0.38 & -0.38 & 0.38 & 0.38 & -0.38 \\ -0.92 & -0.92 & 0.92 & 0.92 & 0.92 & 0.92 & -0.92 & -0.92 \\ 0.54 & -0.54 & 0.54 & -0.54 & 0.54 & -0.54 & 0.54 & -0.54 \end{bmatrix}$$

$$\mathbf{B}_{16Thr,5}(9:16) = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & -1 & -1 & -1 & 1 & 1 & 1 & 1 \\ -1 & 1 & 1 & -1 & 1 & -1 & -1 & 1 \\ 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

3.5.4 Highly Redundant Solution with 24 Thrusters

By increasing the number of thrusters again, we can further improve the redundancy, the controllability envelope and the fuel distribution. Now three thrusters are located on the corners, while the configuration is still geometrically symmetric to the Cartesian plane. This time however no elaborate search of candidate configurations is conducted, but we will directly select the most obvious candidates using the insights obtained from before.

We select two specific configurations, identified by the the stacked orientation vector $[\gamma_1\ \chi_1\ \gamma_9\ \chi_9\ \gamma_{17}\ \chi_{17}]$ for the thrusters located at r_1

$$\begin{bmatrix} 0 & -180 & 0 & -90 & -90 & 0 \\ 0 & -157.5 & -45 & -135 & 0 & -112.5 \end{bmatrix}$$

The other thruster orientations are generated due to mirroring the three presented ones on the planes spanned by the Cartesian axes.

The first configuration results in a mapping matrix $\mathbf{B}_{24Thr,6}$, which is analog to the one used in Subsection 2.5.2, but only with a doubled side-length so that $d = 1m$. This configuration is further decoupled, as now only three entries in the wrench are unequal to zero. The firing directions are now aligned with the respective axis of the geometrical frame. For the right distance to the origin, this type of configuration would result in a homogeneous actuation capability for all forces and torques. For the given side-length we will be able to actuate higher torque values compared to the force values.

The second orientation combination has a more customized distribution of the wrench envelope. The wrench parts f_z and τ_z are less extensive, but in return the other ones can be actuated better. This results in the following mapping matrix

$$\mathbf{B}_{24Thr,7}(1:8) = \begin{bmatrix} -0.92 & -0.92 & 0.92 & 0.92 & -0.92 & -0.92 & 0.92 & 0.92 \\ -0.38 & 0.38 & 0.38 & -0.38 & -0.38 & 0.38 & 0.38 & -0.38 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.38 & -0.38 & -0.38 & 0.38 & -0.38 & 0.38 & 0.38 & -0.38 \\ -0.92 & -0.92 & 0.92 & 0.92 & 0.92 & 0.92 & -0.92 & -0.92 \\ 0.54 & -0.54 & 0.54 & -0.54 & 0.54 & -0.54 & 0.54 & -0.54 \end{bmatrix}$$

$$\mathbf{B}_{24Thr,7}(9:16) = \begin{bmatrix} -0.50 & -0.50 & 0.50 & 0.50 & -0.50 & -0.50 & 0.50 & 0.50 \\ -0.50 & 0.50 & 0.50 & -0.50 & -0.50 & 0.50 & 0.50 & -0.50 \\ -0.71 & -0.71 & -0.71 & -0.71 & 0.71 & 0.71 & 0.71 & 0.71 \\ -0.21 & 0.21 & 0.21 & -0.21 & 0.21 & -0.21 & -0.21 & 0.21 \\ 0.21 & 0.21 & -0.21 & -0.21 & -0.21 & -0.21 & 0.21 & 0.21 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\mathbf{B}_{24Thr,7}(17:24) = \begin{bmatrix} -0.38 & -0.38 & 0.38 & 0.38 & -0.38 & -0.38 & 0.38 & 0.38 \\ -0.92 & 0.92 & 0.92 & -0.92 & -0.92 & 0.92 & 0.92 & -0.92 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.92 & -0.92 & -0.92 & 0.92 & -0.92 & 0.92 & 0.92 & -0.92 \\ -0.38 & -0.38 & 0.38 & 0.38 & 0.38 & 0.38 & -0.38 & -0.38 \\ -0.54 & 0.54 & -0.54 & 0.54 & -0.54 & 0.54 & -0.54 & 0.54 \end{bmatrix}$$

3.6 Comparison of Obtained 3D-Configurations

This section discusses the performance of the previously introduced subset of candidate configurations. Potentially, there can be found infinitely many other reasonable ones. The following results are also not described to their full extent, as they match with the previously obtained insights.

It is difficult to compare the performance of different configurations with a different number of thrusters and different design goals, but once again we will try to do so with the indexes developed in the beginning of this section. It also has to be stressed again that, due to the use of the Euclidean distance, the indexes in the current form heavily favor more isotropic configurations.

In conclusion, we will observe that a large amount of well-placed thrusters improve the shape of the controllability envelope and thus also the performance of the configuration. As many thrusters as possible should be used in order to be able to generate a homogeneous behavior. This also allows to cope with malfunctioning thrusters.

Form-Closure and Manipulability With the help of these indexes a pre-selection of the larger set of candidate configurations was done. The fixed locations of the thrusters prohibited to obtain $I_m = 1$ (see Table 3.3).

Only the Pythagorean solution $\mathbf{B}_3$ is isotropic in all wrench dimensions with the singular values all equal to two. For this candidate however no $I_{m,f}$ and $I_{m,\tau}$ can be given though, as the left singular vectors of the SVD are not equal to signed unit vectors. We can also observe in the table that an increase in thrusters automatically increases the average magnitude of the singular values.

Control Capability The increase in the magnitude of the singular values is also reflected in the control capability. A large amount of thrusters can cover a larger wrench space.

In the results of configuration $\mathbf{B}_5$, we can see how an adapted weighting of the wrench dimensions affects the distance measure. The envelope boundary is heavily skewed and has a large deviation between minimal and maximal distance to the origin.

Another notable observation can be made when comparing $\mathbf{B}_6$ and $\mathbf{B}_7$. Both use a high amount of thrusters, but $\mathbf{B}_6$ seems to outperform $\mathbf{B}_7$, not only in the controllability but also other parameters. Even though heavily coupled actuation wrenches are required for a small amount of thrusters, with more thrusters in the configuration, a more decoupled configuration should be used.

	I_c	I_m	$I_{m,f}$	$I_{m,\tau}$	$I_{d,min}$	$I_{d,max}$	$I_{d,avg}$	$I_{d,std}$	I_v
$\mathbf{B}_1$	1	0.33	2.59	0.86	6.06	27.17	9.64	2.10	8.09e6
$\mathbf{B}_2$	1	0.20	1.59	1.84	5.52	29.99	9.58	2.59	1.04e7
$\mathbf{B}_3$	1	1	/	/	14.15	33.40	20.51	2.90	4.90e8
$\mathbf{B}_4$	1	0.71	11.31	11.31	20.03	52.10	29.10	3.89	3.88e9
$\mathbf{B}_5$	1	0.28	8.00	17.85	12.55	72.01	26.44	8.99	6.12e9
$\mathbf{B}_6$	1	0.71	22.63	64	40.05	86.33	58.34	7.43	2.41e11
$\mathbf{B}_7$	1	0.63	20.00	18.06	28.30	68.84	49.07	5.75	7.84e10

Table 3.3: Part 1: Index values for three-dimensional configurations

Fuel Distribution The significance of using more thrusters with decoupled wrenches can also be seen in the fuel consumption. We recall that three different scenarios were generated in order to measure the fuel distribution of comparable control wrenches. The first scenario uses a complete wrench with a radius equal to u_{max}. This radius is generated by calculation of the L2-norm of the sampled wrench. Scenario two and three sample from the subset of clean forces and torques while using the same radius.

Mainly two outcomes of this observation have to be noted. First, unsurprisingly, more available thrusters reduce the average amount of fuel, which is required, depending on the scenario, for a random wrench, force or torque with constant radius. As noted in the previous sections, wrenches which correspond to the actuation of one single thruster can be actuated very efficiently. We have weak directions for which we have to combine a large amount of thrusters and where certain parts of their wrenches are counter actuated. For more thrusters, the single thrust vectors are distributed better in the complete wrench space and thus minimize the effect of the weak directions.

Second, we can see that even though $\mathbf{B}_3$ uses only 12 thrusters overall, which naturally leads to a reduced envelope, this configuration is capable to minimize the deviation of fuel consumptions. For all scenarios (see tables 3.3 and 3.4) the fuel consumption on average is comparable to configurations with more thrusters.

On a side-note, it has to be added that the fuel index results for $\mathbf{B}_1$ and $\mathbf{B}_2$ are only given for wrenches which are inside the controllability envelope. Due to the limited amount of thrusters, a wrench with the selected radius cannot be actuated in every direction. This means that depending on the scenario up to two thirds of the sampled wrenches become infeasible.

This problem once again stresses the difficulties we have with finding neutral indexes that allow us to compare different configurations. Clearly, depending on the mission requirements of the spacecraft, the distance measure has to be adapted. Otherwise, as also visible for the results in $\mathbf{B}_5$, the indexes might be deceptive.

	$I_{f1,min}$	$I_{f1,max}$	$I_{f1,avg}$	$I_{f1,std}$	$I_{f2,min}$	$I_{f2,max}$	$I_{f2,avg}$	$I_{f2,std}$
$\mathbf{B}_1$	10.00	51.63	29.62	6.64	12.47	40.00	29.51	7.01
$\mathbf{B}_2$	8.82	55.25	28.58	6.51	11.58	40.00	28.49	6.70
$\mathbf{B}_3$	7.47	17.32	14.41	1.28	13.76	16.18	15.00	0.59
$\mathbf{B}_4$	7.62	19.97	14.63	1.98	10.00	20.00	15.24	2.60
$\mathbf{B}_5$	6.16	35.18	20.31	5.76	10.00	30.00	23.48	5.09
$\mathbf{B}_6$	5.97	17.30	11.51	1.95	10.00	17.32	15.00	1.53
$\mathbf{B}_7$	7.20	23.24	13.66	2.64	10.75	14.55	12.44	0.94

Table 3.4: Part 2: Index values for three-dimensional configurations

	$I_{f3,min}$	$I_{f3,max}$	$I_{f3,avg}$	$I_{f3,std}$	I_r
$\mathbf{B}_1$	15.40	52.24	34.08	9.61	0.00
$\mathbf{B}_2$	11.83	44.30	31.27	7.79	0.00
$\mathbf{B}_3$	13.76	16.18	15.00	0.59	0.00
$\mathbf{B}_4$	10.00	20.00	15.24	2.61	1.80
$\mathbf{B}_5$	7.00	20.00	13.90	3.62	1.967
$\mathbf{B}_6$	7.07	10.00	8.66	0.65	3.997
$\mathbf{B}_7$	8.82	18.48	12.57	2.33	3.9998

Table 3.5: Part 3: Index values for three-dimensional configurations

Level of Redundancy The good distribution of thrusters in $\mathbf{B}_3$ comes at the cost of having no redundancy even though significantly more than the minimal amount of seven thrusters are used. This means that there also is a trade-off between the level of redundancy and the other performance parameters for configurations with the same amount of thrusters. Our two configurations with 24 thrusters almost allow that four random thrusters can break down at the same time. However, we can find a few combinations where form-closure is lost, so that we have $I_r < 4$.

One open question, with respect to this index, is how relevant it is. Currently it is only verified that form-closure is kept. However, we have no idea how well certain wrench directions can be actuated in the worst case, after multiple thruster failures, without further investigation.

4

Control of a One-Dimensional Point Mass

It does not suffice to merely model the thruster actuation from a simple input-output point of view like in Chapter 2. The thrusters have to be used together with the system and a controller in a closed loop in order to learn about their dynamical behavior.

As a first step, we will utilize thrusters in a very simplistic one-dimensional environment and reduce the system to a point mass. We will formulate certain characteristics of the controlled thrusters, which can be also observed in multi-dimensional cases. Additionally we will impose criteria which can be used to tune the parameters of the controller. It will become obvious that the used approach for the thruster assignment has a direct influence on the tuning of the controller.

For a one-dimensional task only two thrusters, oriented in the opposite direction, are required. The thruster assignment problem is trivial and only consists of selecting the thruster corresponding to the correct sign of the control force.

4.1 D-Control with Pulse-Width Modulated Thruster

In an environment without friction, as present in space, an uncontrolled velocity will lead to an infinite drift of the system. Thus, this control task consists of reducing an initial velocity of the point mass to zero. We will be able to see one first key limitation, which comes from using thrusters as actuator. The residual drift cannot be eliminated, but only reduced up to a certain minimal bound, which is dependent on the thruster properties and actuation.

4.1.1 Ideal Continuous D-Control

A one-dimensional D-control generates a force dependent on a viscous damper with $K_d \in \mathbb{R}^{1x1}$ and a velocity difference

$$F_D = K_D \cdot (\dot{x}_{des}(t) - \dot{x}(t)) = K_D \cdot \Delta v(t)$$

with x being the position and $\dot{x}$ the velocity of the body.

Note that the force is generated in the opposite moving direction. For simplicity, we will from now on denote the velocity with $v(t) = \dot{x}(t)$ and the acceleration with $a(t) = \ddot{x}(t)$ within this chapter.

Figure 4.1 depicts the simple closed loop system for the 1D point mass assuming an ideal force actuation.

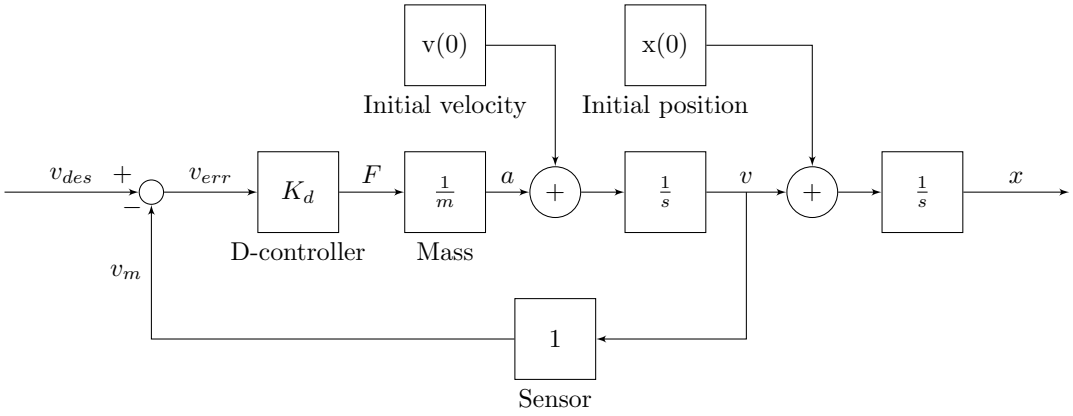


Figure 4.1: Continuous D-control scheme

Here, the control behavior can be easily described with the help of a Laplace transform

$$V(s) = \frac{\frac{K_d}{m}}{s + \frac{K_d}{m}} \cdot V_{des} + \frac{1}{s + \frac{K_d}{m}} \cdot V(0) \tag{4.1}$$

and

$$X(s) = (V(s) + X(0)) \cdot \frac{1}{s} \tag{4.2}$$

using the Laplacian rules for derivatives $\mathcal{L}(\ddot{x}) = A(s) = s^2X(s) - s \cdot X(0) - V(0)$ and $\mathcal{L}(\dot{x}) = V(s) = sX(s) - X(0)$. As we have a constant desired velocity $v_{des} = 0$, only the initial velocity indicated by $V(0)$ is of relevance.

Transforming the Laplace formulation back to time domain, a continuous D-control without limitations on the applicable force will change the velocity of the mass according to

$$v(t) = V(0) \cdot e^{-\frac{K_D}{m}t} \tag{4.3}$$

and accordingly the position will change

$$x(t) = V(0) \cdot \frac{m}{K_D} - V(0) \cdot \frac{m}{K_D} \cdot e^{-\frac{K_D}{m} \cdot t} \tag{4.4}$$

Both equations were obtained by generating transformable formulations with the help of partial fraction expansion. These equations will be used in the following in order to compare the discrete thruster-controlled system to this ideal performance.

4.1.2 Bound for Discrete Control

The real system however will be discretely controlled. This means that the control output will be only time-discretely updated which will result in step-like control signals. Though, the discrete version behaves very similar to the continuous controller as long as the sampling time, here the inverse of f_{ctrl}, is sufficiently small.

Discretization effects or an unstable system may be observed until sufficiently high control frequencies are used. After this point, a quasi-continuous behavior is observed. The area, where this is the case, is typically only given heuristically. The basis are different parameters of both the analogous closed loop and the open loop system. Unbehauen [67] states that the control time interval should be 0.1 times the most dominant time constant. The control frequency should be the 20-fold of the bandwidth of the closed control frequency according to [68], while in [69] the 6- to 20-fold of the biggest frequency in the control loop is deemed sufficient. Zacher [70] comes to similar conclusions for a wide range of possible system characteristics.

As visible in equation 4.3 and the corresponding Laplace formulation, the only time constant of the controlled system is given by

$$\tau_d = \frac{m}{K_d} \tag{4.5}$$

We can use this knowledge to derive a boundary condition for an asymptotically stable solution. Assuming no disturbances so that the system is correctly expressed by 4.1, we can guarantee a convergence for

$$f_{ctrl} \geq \frac{1}{\tau_d} \tag{4.6}$$

This means selecting $f_{ctrl} = \frac{1}{\tau_d}$ represents the lowest possible control frequency which can be used for a given time constant so that the control behavior is definitely asymptotically stable. We saw before that the time constant is dependent on the selected K_d, which is the only open parameter on the controller side. We can reformulate this to get a formulation for the highest gain

$$K_{d,max} = f_{ctrl} \cdot m \tag{4.7}$$

By selecting this gain, we theoretically converge after one control time interval. This can be seen by differentiation of equation 4.3 at time zero. We apply $a(0) = -\frac{K_d}{m} \cdot V(0)$ for a complete control time interval, which results in a zero velocity after exactly one time interval.

Note that we can still converge to stable solutions for larger values of $K_d > K_{d,max}$, as we will see later on, due to the nature of the thrusters. Generally though, in order to be robust against uncertainties and have a quasi-continuous control, we want to have $K_d < K_{d,max}$.

We can reformulate from finding a suitable value for the differential gain K_d to selecting a good coefficient

$$c_{samp} = \frac{f_{ctrl} \cdot m}{K_d} \tag{4.8}$$

with the condition of $c_{samp} \geq 1$.

This coefficient embeds the previously developed notions of selecting the correct f_{ctrl} and K_d. A higher value for c_{samp} will result in a closer following of the e-function given by equation 4.3.

4.1.3 Time-Discrete D-Control with Thrusters

Recall of Definitions In Subsection 2.4.2 some key equations were developed, which we will use in the following. Equation 2.50 describes which values of assigned thrust can be realized for a thruster. Equation 2.51 gives a limit on the usable control frequency and 2.52 indicates how the pulse lengths are discretized.

The left side of the plot in Figure 4.2 gives an overview how an arbitrary assigned thrust, denoted by $u_{thr,sel}$, is discretized by the PWM in thrust space, $u_{thr,act}$, for a given control time interval. The dead-zone u_{dead} can be observed and manifests itself in a discretization gap. For two examples (red and blue), we can see on the right side, how the discretized thrust assignment is translated into a pulse, specified in the time domain.

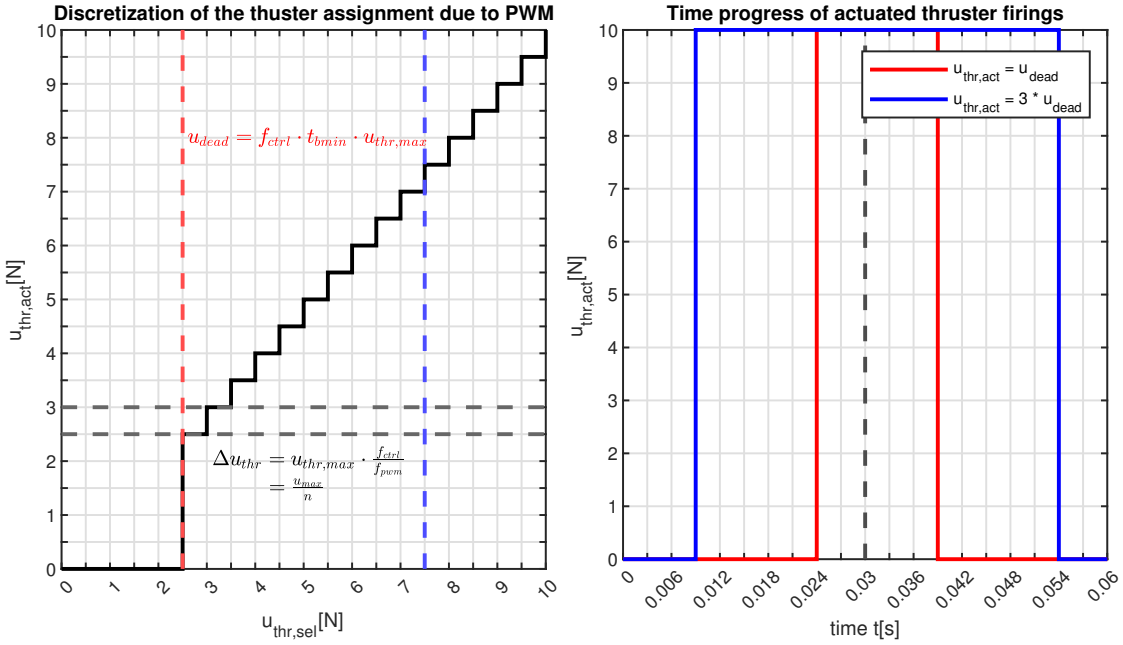


Figure 4.2: Realization of the thruster firings for a PWM with $t_{mib} = 0.015s$, a control frequency of $\frac{1}{4 \cdot t_{mib}} = 16.67Hz$ and $f_{PWM} = 5 \cdot \frac{1}{t_{mib}} = 333Hz$

Note that the grid is selected in a way so that it matches the PWM time intervals. In the given example, an assignment of u_{dead} equals five PWM time intervals. This is also why the pulse is not ideally centered, as this is only possible for even numbers.

Minimal Change in Velocity One very important characteristic system parameter, the minimal change in velocity, identifies how this system state changes if one thruster is firing for the minimal valve opening time t_{mib}. In the three-dimensional setting, depending on their respective mounting, each thruster will affect different parts of the 6-dimensional velocity vector containing both rotational and translational velocities.

This parameter can be easily formulated for the one-dimensional case and the consequential thruster configuration, as there exist no couplings across different dimensions

$$\Delta v_{mib} = \frac{u_{max}}{m} \cdot t_{mib} \tag{4.9}$$

Simple Thruster Assignment In the case of only two thrusters in a one-dimensional setting, we have a straightforward assignment when we ignore t_{mib}. All negative forces have to be realized by one thruster, the positive ones by the other one. We have

$$\mathbf{B} = [1 \ -1] \tag{4.10}$$

and can equate

$$\mathbf{u} = \begin{cases} [u_{max} \quad 0], & \text{for } u_{max} < F_{ctrl} \\ [F_{ctrl} \quad 0], & \text{for } 0 < F_{ctrl} < u_{max} \\ [0 \quad 0], & \text{for } F_{ctrl} = 0 \\ [0 \quad |F_{ctrl}|], & \text{for } -u_{max} < F_{ctrl} < 0 \\ [0 \quad u_{max}], & \text{for } F_{ctrl} < -u_{max} \end{cases} \tag{4.11}$$

This simple assignment is the one we would obtain from the standard LP formulation as stated in equation 2.12. Note that no specific notion of u_{dead} is included and thus assignments $u < u_{dead}$ will be set to zero during the PWM.

Control Tuning When using thrusters, it is impossible to perfectly reduce the final velocity to zero, with exception a small subset of initial velocities. We can use equation 2.50 to relate the maximally tolerated final velocity v_{fin} which results in a firing of a thruster with the minimal change in velocity.

When using the simple assignment logic, the system will only counteract velocities which lead to a demanded thrust higher than the natural dead-zone

$$|K_d \cdot v_{fin}| = f_{ctrl} \cdot t_{mib} \cdot u_{max} \tag{4.12}$$

By reformulating K_d as a parameter of c_{samp}, as given by equation 4.8, we can obtain a relation which links our developed notion of the minimal change in velocity to v_{fin}.

$$|v_{fin}| = c_{samp} \cdot \Delta v_{mib} \tag{4.13}$$

This allows to develop an intuition of the minimal value of c_{samp} from a new point of view, as it directly affects the control performance. v_{fin} is the velocity that triggers a firing that results in a change in velocity of Δv_{mib}. For the system in order to be stable, we can only demand a $v_{fin} > \frac{\Delta v_{mib}}{2}$ or equally $c_{samp} \geq 0.5$.

For different selections of c_{samp}, we can identify three different ranges of operation:

- $c_{samp} \geq 1$: The final velocity will have the same sign as the initial velocity and will be located within the bound given by $\begin{bmatrix} 0 & v_{fin} \end{bmatrix}$
- $0.5 < c_{samp} < 1$: The converging velocity may change sign compared to the initial velocity and will be located within the bound given by $\begin{bmatrix} -v_{fin} & v_{fin} \end{bmatrix}$
- $c_{samp} < 0.5$: Discretely controlled system does not converge to a solution and may even become unstable

First Simulation The previously obtained results are verified with the help of a simulation. The initial velocity was specified to be $v(0) = \frac{2}{30}\frac{m}{s}$, while we have $m = 170kg$ for the point mass. The thruster specification can be given by $t_{mib} = 0.015s$ and a nominal thrust $u_{max} = 10N$.

We can identify that there are three open parameters left, whose effect on the control behavior will be analyzed. The frequency of the PWM f_{pwm} affects the discretization error of the force during the modulation. Both the control frequency f_{ctrl} and the constant c_{samp} specify the control gain (by using equation 4.8) and also the final velocity (see equation 4.13).

We want to focus first on how the velocity changes over time in the following. In the figures 4.3, 4.4 and 4.5, this property is plotted for a selected set of control frequencies with different simulation setups. In each plot, a different value for c_{samp} and f_{pwm} is fixated. Note that the dotted lines indicate the theoretical result of the analogous continuous control with an ideal actuator.

By selecting $c_{samp} = 10$, we specified a quasi-continuous behavior, which can also be observed in the simulation, as visible in Figure 4.3. With our parameter tuning approach, we independently specify the same maximally tolerated final velocity v_{fin} for all possible f_{ctrl}. Some results are given in the figure and we can see that the velocity of the controlled system will be always lower than v_{fin}. As noted in Subsection 2.4.2, the control frequency $f_{ctrl} = 66.67Hz$ can be seen as a form of Bang-Bang modulation. However, as the control gain is also affected by the control frequency, the convergence rate, specified by the time constant of the system, is different for each f_{ctrl}. Additionally, we can observe that f_{pwm} hardly changes the performance. The boundedness of the thrust strength results in a maximal slope of the velocity, which gets apparent, for example, for $f_{ctrl} = 33.33Hz$ in the plot. The convergence slope for the unbounded continuous controller (dotted lines) cannot be followed.

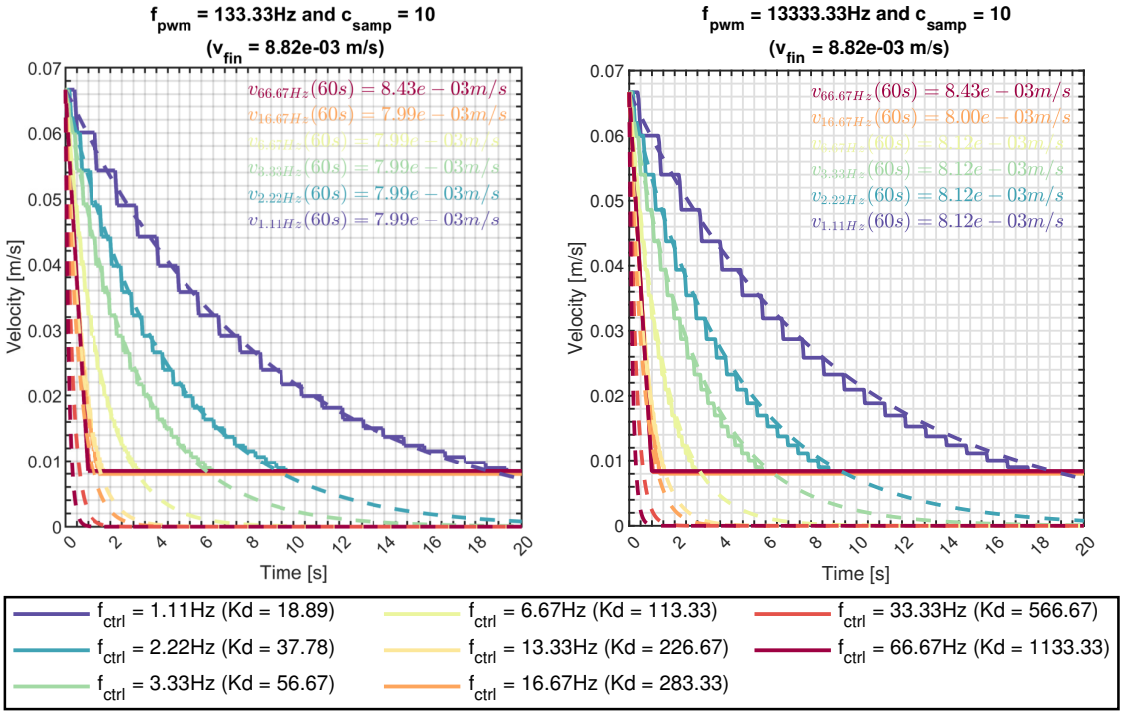


Figure 4.3: D-Control (simple assignments): Time response of velocity ($c_{samp} = 10$)

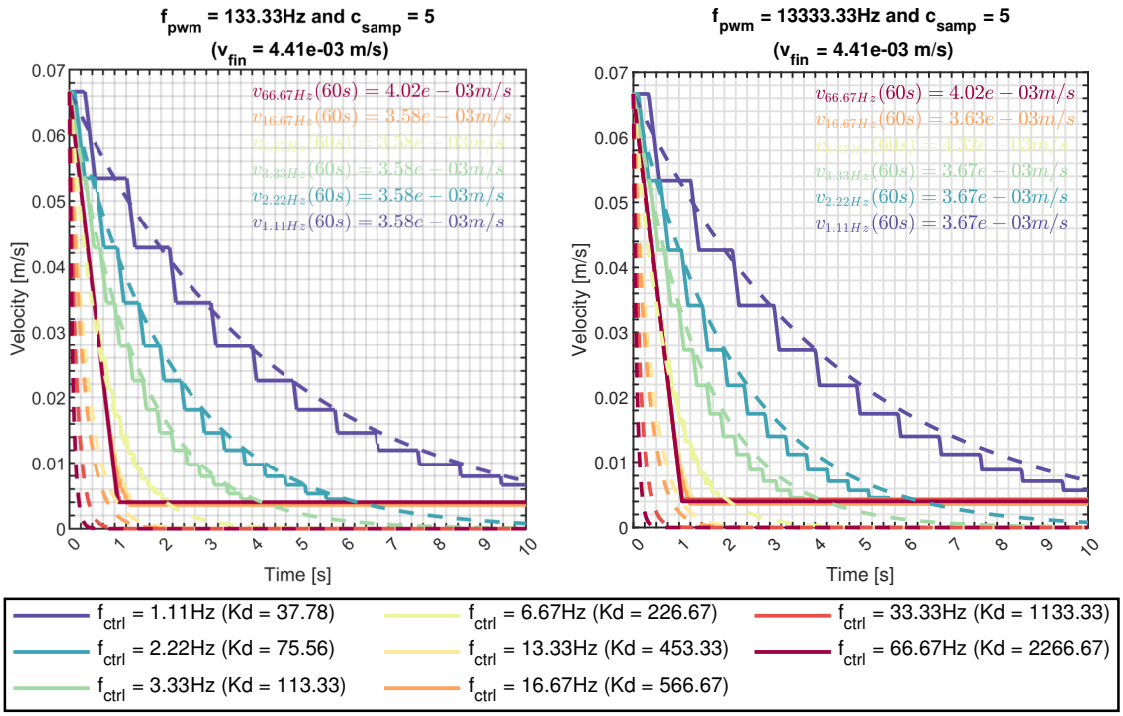


Figure 4.4: D-Control (simple assignments): Time response of velocity ($c_{samp} = 5$)

By reduction of c_{samp}, as visualized in Figure 4.4, the effects of the discretization of the continuous controller gets more apparent. Additionally, we can see that stricter bounds for v_{fin} can be achieved. The actuator saturation leads to a larger discrepancy between the desired curves of the e-functions, given by the continuous control with ideal actuators, and the discretely controlled system with bounded thrusters.

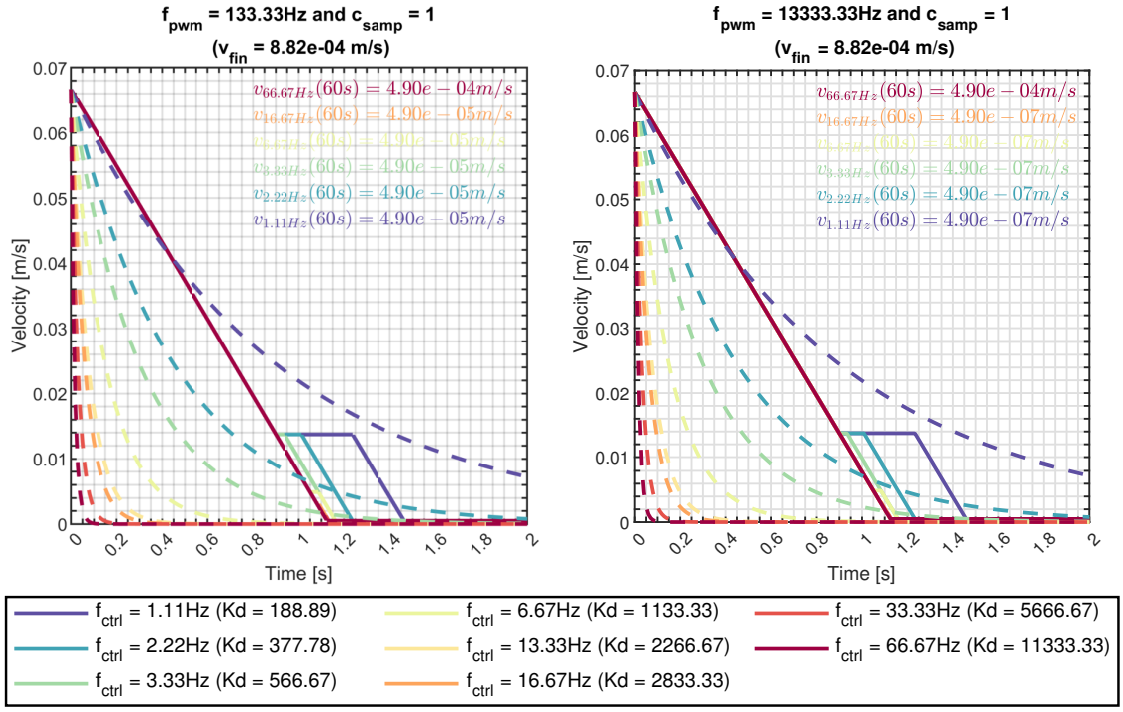


Figure 4.5: D-Control (simple assignments): Time response of velocity ($c_{samp} = 1$)

Using the simple thruster assignment method, the lowest possible value for v_{fin} without a change in the sign of the velocity can be specified by $c_{samp} = 1$, as depicted in Figure 4.5. We can see that in this case the thrusters are saturated for all control time intervals apart from the last one before convergence. Depending on the remaining velocity within this last time interval, an according pulse length will be given, which guarantees no firing afterwards.

With the given selections of c_{samp}, an investigation of the fuel consumption is unnecessary. It can be directly derived from the difference of initial and final velocity, as only one thruster is used during the deceleration task. No fuel is wasted and only used to achieve the desired task of slowing down the system.

Sophisticated Thruster Assignment We saw that the selection of the gain can be directly linked to the final velocity bound. With our simple method of determining thruster assignments, the final velocity is thus limited to the system parameter of the minimal change in velocity.

Smaller velocities could only be achieved, if we were able to realize smaller relative pulse lengths.

Subsection 2.3.4 abstractly introduced the notion of counter-thrust, which can be trivially realized in our simplified one-dimensional setting. In our simple assignment we had no knowledge of the dead-zone, which led to the system being not able to actuate below this threshold. A velocity of zero could not be achieved apart from some special cases. However, with the help of the antagonistic second thruster, we can in fact realize smaller relative pulse-lengths and therefore theoretically achieve a velocity of zero.

The following assignment logic is equal to the counter-thrust notion.

$$\mathbf{u} = \begin{cases} [u_{max} \quad 0], & \text{for } u_{max} < F_{ctrl} \\ [F_{ctrl} \quad 0], & \text{for } u_{dead} < F_{ctrl} < u_{max} \\ [F_{ctrl} + u_{dead} \quad u_{dead}], & \text{for } 0 < F_{ctrl} < u_{dead} \\ [0 \quad 0], & \text{for } F_{ctrl} = 0 \\ [u_{dead} \quad |F_{ctrl}| + u_{dead}], & \text{for } -u_{dead} < F_{ctrl} < 0 \\ [0 \quad |F_{ctrl}|], & \text{for } -u_{max} < F_{ctrl} < -u_{dead} \\ [0 \quad u_{max}], & \text{for } F_{ctrl} < -u_{max} \end{cases} \tag{4.14}$$

Analogously, by replacing the zeros in the logic conditions above with u_{disc}. Therefore, once again we prevent actuated thrust pulses of the same length for both thrusters which completely cancel each other out. This step thus includes the limitations on the minimally achievable relative thrust due to the limited resolution of the PWM. Note that, with our simple configuration, the resulting thruster assignment is also equal to the solution we would obtain with a MILP formulation, as introduced in Section 2.3.3. However, this does not hold generally.

We can reformulate equation 2.53 for u_{disc} to stress its usefulness in expressing the thrust resolution of the PWM in thrust space.

$$u_{disc} = \frac{f_{ctrl}}{f_{PWM}} \cdot u_{max} = \frac{u_{dead}}{t_{mib} \cdot f_{pwm}} \tag{4.15}$$

With our new assignment, this parameter can be interpreted as a new dead-zone for $f_{pwm} \geq \frac{1}{t_{mib}}$. All assignments above this value will lead to a firing of the thrusters.

More intuitively, one can show this with an example, like given in Figure 4.6. Our new assignment method translates every assignment below u_{dead} to a feasible, proportional one above u_{dead} through addition of this value. To keep the same relative thrust from before, we have to add u_{dead} also to its antagonistic thruster.

With our one-dimensional system the identification of the antagonists is trivial. We can also see in the figure, why we should only transform assignments above u_{disc}. Otherwise after the PWM, we would actuate both thrusters with u_{dead} and thus achieving nothing except from wasting fuel.

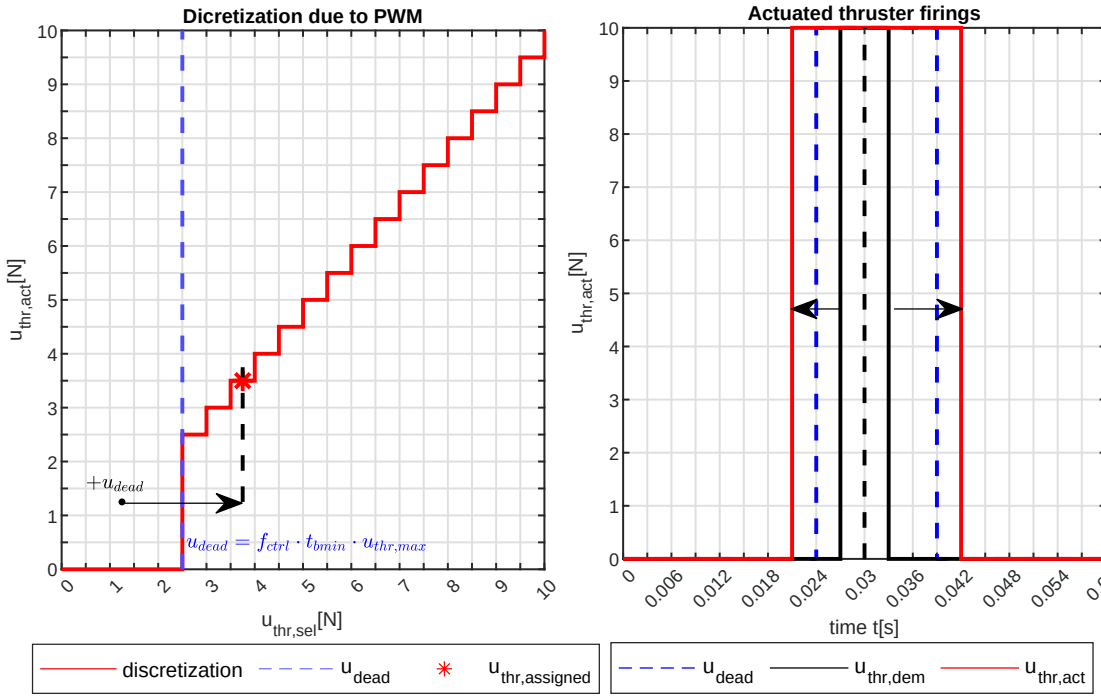


Figure 4.6: Realization of the counter-thrust firing for an assigned thrust of $u_{thr,dem} = 1.25N$ with $t_{mib} = 0.015s$, a control frequency of $\frac{1}{4 \cdot t_{mib}} = 16.67Hz$ and $f_{pwm} = 5 \cdot \frac{1}{t_{mib}} = 333Hz$

Additionally, we can see where the condition in equation 2.38 comes from. No distortion of the relatively applied force due to actuator saturation is committed as long as there are as many assignments on the right side of u_{dead} as on the left. One can express this condition also with respect to the maximally possible control frequency which should be used with the counter-thrust notion.

$$f_{ctrl} \leq \frac{1}{2 \cdot t_{mib}} \tag{4.16}$$

Second Simulation Due to the more sophisticated assignment logic, the convergence velocity can now be tuned by the selection of f_{pwm}. We are not bounded by a fixed system property. The minimal change in velocity due to the smallest relative pulse is now given by

$$\Delta v_{pwm} = \frac{u_{max}}{m} \cdot \frac{1}{f_{pwm}} \tag{4.17}$$

Note that this expresses an altered minimum change in velocity and we express this by a new symbol in order not to mix up these two. The bound for the final convergence velocity is now given by

$$|v_{fin}| = c_{samp} \cdot \Delta v_{pwm} \tag{4.18}$$

This parameter is still dependent on the selected differential gain and the control frequency but can be now additionally tuned by f_{pwm}.

The progress of the velocity still follows the theoretical e-function as before (see Figure 4.7). Only now the system can achieve lower velocities. For an infinite value of f_{pwm}, in theory a final velocity of zero could be achieved. The figure provides us also with a numerical comparison of the quality of the final velocity, which is significantly lower for the more sophisticated, second assignment method used on the right side. While the left plot cuts off the e-function at a relatively high velocity, it is followed longer, until almost zero, on the right side. Apart from this fact, the shape and slope of the plots do not change, so that no additional plots are provided. Note that the small gain for $f_{ctrl} = 1.11Hz$ means that v_{fin} is still not reached after 60s.

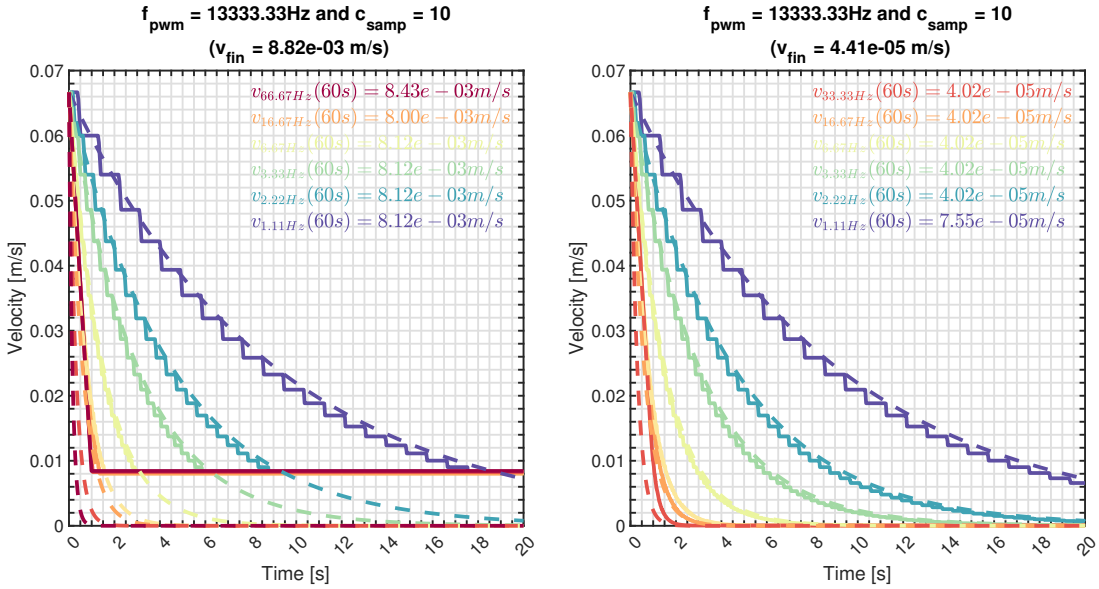


Figure 4.7: D-Control with simple (left) and advanced (right) assignments: Time response of velocity ($c_{samp} = 10$); legend for f_{ctrl} is the same as 4.3

Another interesting property, namely the fuel consumption, comes up within the context of using multiple thrusters to generate smaller relative pulses. Obviously, more fuel will be required for the second assignment method. As the two thrusters work against each other, we can say that fuel is wasted during the process.

It has to be noted that, during this work, we somehow sloppily exploit the term of fuel consumption. Previously, we just assumed that the summed up thrust assignments can be simply multiplied with a constant factor in order to derive a proper fuel consumption during one control time interval. In the closed loop scenario, we change this notion slightly. Now we integrate the applied thrusts of all firing thrusters over time in order to denote the fuel consumption.

Therefore, the different kinds of fuel consumptions across the different chapters are not directly comparable. In the given setup of Figure 4.7, the overall fuel consumption, for example, for $f_{ctrl} = 6.67Hz$ is at around $10Ns$ for the first assignment type and at $28Ns$ for the second one. A fuel consumption of $10Ns$ equals one thruster firing for one second. This means that in order to achieve the lower final velocity, the cumulated thruster firing time is almost tripled.

We would like to get a better understanding how we can affect the fuel consumption here. Instead of considering control forces and thruster assignments, we can consider firing times to motivate this discussion. In order for our simple system to stop an initial velocity, we have to fire the thruster which is located in the opposite moving direction for a specific amount of time given by

$$t_{fire} = V(0) \cdot \frac{m}{u_{max}} \tag{4.19}$$

For our simulation experiment this equals to a fuel consumption of around $11.3Ns$. Depending on the gain, this firing time is distributed across the different control time intervals. After each control time interval, we have a remaining firing time which can be calculated from the residual velocity. The first assignment method made sure that only the one required thruster fires. However, it cannot reach a velocity of zero. It will lead to a convergence as soon as its assigned firing time goes below t_{mib} during one control time interval. The remaining velocity can be calculated from the sum of preceding firings. With our second assignment method, we can realize very small effective pulse lengths, but in order to achieve these, both thrusters have to fire for an additional time t_{mib}.

Effectively, only a fraction of the combined firing time is used to slow down the system with a relative thrust u_{rel}, which makes this operation quite costly from the perspective of fuel consumption. This means the efficiency bounds of the counter-thrust operation can be given by

$$0\% < \frac{u_{rel}}{u_{rel} + 2 \cdot u_{dead}} < 33.33\% \tag{4.20}$$

The solution space of possible relative pulse lengths is divided across u_{dead}. On the left side, we have costly assignments which will waste fuel. These were not available with the first assignment method. On the right side, we have no fuel waste.

When recalling the behavior and dependency from c_{samp} for the closed loop system, we can reason that lower values for c_{samp} will lead to less fuel waste. We noted before that for $c_{samp} = 1$, the thrusters continuously fire for all control time intervals except from the last one before convergence. Only during this control time interval counter-thrust might be used to achieve the demanded relative thrust. For random initial velocities, the probability to use counter-thrust at all can be given by

$$p_{counter} = \frac{u_{dead}}{u_{max}} \tag{4.21}$$

It becomes apparent that the chance of using counter-thrust is very much dependent on the used f_{ctrl}. If, for a given initial velocity and $c_{samp} = 1$, we are not requiring counter-thrust in the last control step before convergence, the first assignment method would have given us the same final velocity. Larger values for c_{samp} will lead to higher probabilities to not only require additional fuel in the last time interval before convergence, but also before, as the slope of the e-function is less steep.

Conclusion To conclude, we can see that the thruster assignment logic can influence our very simple controllability envelope given by the two thrusters. Inherently, some control forces are not feasible due to the thruster configuration and the limit on the maximal thrust, which is specified by the controllability envelope. However, depending on the assignment, we can also have controllability holes inside. Here, some control forces will be mapped to infeasible assignments, which have to be corrected and will realize differing forces.

In both cases we have to map to feasible control forces. Also, we could see that even for only two thrusters, the fuel consumption is not proportional to the demanded force, but irregularly distributed. This also verifies that the control frequency and the control gains directly affect this parameter.

4.2 PD-Control with Pulse-Width Modulated Thruster

Our second simplified control task consists of following a given 1D-trajectory and holding the final position in the end. Still, a mass of $m = 170kg$, which is actuated by two opposing thrusters, is used within the simulation.

We want to analyze how the system is now affected by different selections of the proportional gain K_p and how it can be tuned depending on the properties of the used thruster. Note that here we are only using the simple thruster assignment method, as described in equation 4.11.

4.2.1 Ideal Continuous PD-Control

Control Scheme The continuous realization of the used control scheme is depicted in Figure 4.8. We will be using a very simple PD-control without a pilot control.

Both, velocity and position of the mass are measured and used to generate a control force depending on their error. Without a feedforwarding part in the control, the quality of the trajectory following is inherently prone to a slope error. As we, however, are primarily interested in the properties of the final position holding, this kind of error can be tolerated.

The generated control force can be seen as a superposition of the two sub-parts

$$F_{ctrl} = F_P + F_D = K_P \cdot (x_d - x) + K_D \cdot (\dot{x}_d - \dot{x})$$

with x_d and $\dot{x}_d$ being the time-dependent, desired states for position and velocity.

The closed loop system can be expressed by a simple spring-mass-damper system. Without further prove, we can relate the two gains in order to critically damp the system by

$$K_d = 2 \cdot \zeta \cdot \sqrt{m \cdot K_p} \tag{4.22}$$

with $\zeta = 1$ being the coefficient for a critically damped system.

This corresponds to the one-dimensional solution of the generalized multi-dimensional case

$$K_d = \zeta(M_1 K_{p1} + K_{p1} M_1) \tag{4.23}$$

with $K_{p1}K_{p1} = K_p$ and $M_1 M_1 = M$ with both the proportional gain K_p and the inertia M being positive definite matrices.

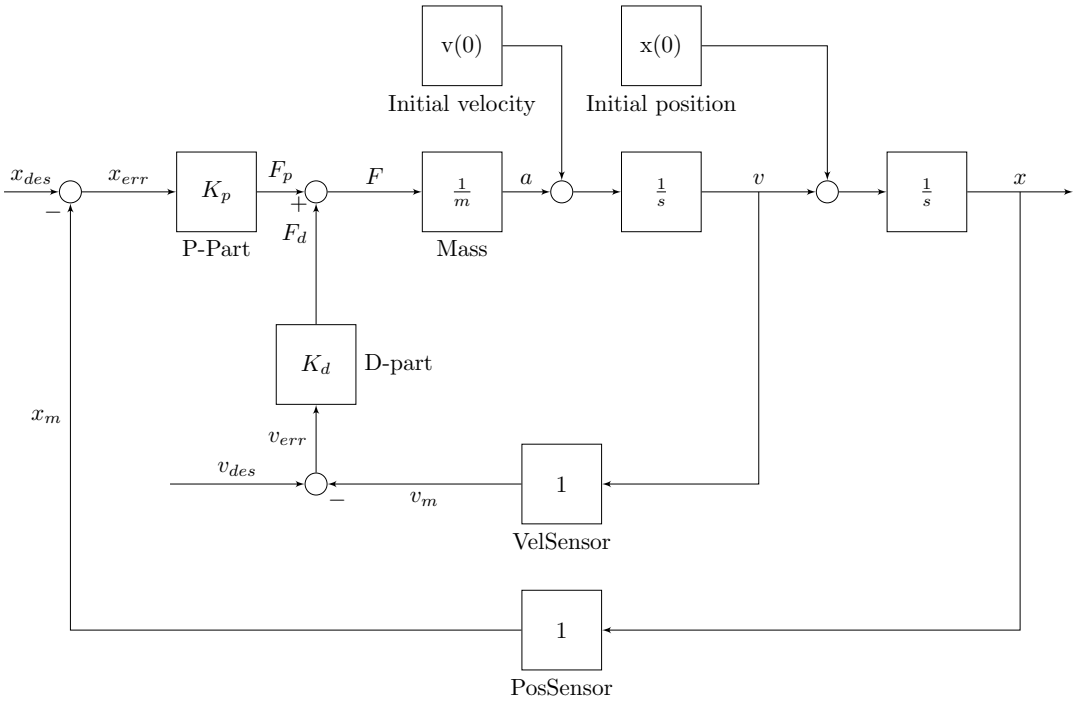


Figure 4.8: Continuous PD-control scheme

From now on, the initial values for position and velocity are set to zero in order to simplify the following calculations. The progress of the position only depends on the desired velocity and position. In Laplace domain, the output, from now on denoted as $X(s)$, can be given with

$$X(s) = X_{des} \cdot G_{PX} + V_{des} \cdot G_{DX} = X_{des} \cdot \frac{K_p}{m \cdot s^2 + K_d \cdot s + K_p} + V_{des} \cdot \frac{K_d}{m \cdot s^2 + K_d \cdot s + K_p} \tag{4.24}$$

with X_{des} and V_{des} being the desired inputs in the Laplace domain. The transfer functions G_{PX} and G_{DX} can be derived by using the separation principle.

The denominator of both fractions also shows why the fixation of K_d given in equation 4.23 makes sense. Then, only real poles are obtained. One can immediately recognize the Laplace transform of the mass-spring-damper system.

Trajectory Generation A trajectory for a given start and end point can be fully specified by a fifth order polynomial, assuming that the start and final velocities and accelerations are also specified. This approach could also be expanded to work in more than one dimension, but is not required here.

$$x_{des}(t) = a_0 + a_1 \cdot t + a_2 \cdot t^2 + a_3 \cdot t^3 + a_4 \cdot t^4 + a_5 \cdot t^5 \tag{4.25}$$

The velocity is specified by the derivative

$$v_{des}(t) = a_1 + 3 \cdot a_2 \cdot t + 3 \cdot a_3 \cdot t^2 + 4 \cdot a_4 \cdot t^3 + 5 \cdot a_5 \cdot t^4$$

and the acceleration is given by the second derivative

$$a_{des}(t) = 2 \cdot a_2 + 6 \cdot a_3 \cdot t + 12 \cdot a_4 \cdot t^2 + 20 \cdot a_5 \cdot t^3$$

By specifying the initial and stopping conditions for position, velocity and acceleration, it is possible to solve for the missing coefficients $a_0 - a_5$ (6 equations for 6 unknowns). Additionally, the points in time of start and stop have to be pre-defined.

Typically the velocities and accelerations are set to be zero at the start and stop of the trajectory. Also without loss of generality, the starting time can be set to zero, in order to simplify the calculations. Thus, only three parameters have to be set to be able to solve for the coefficients: the start position x_s, the end position x_e and the end time t_e.

The coefficients are generally given as dependent variables:

$$a_0 = x_s \tag{4.26}$$
$$a_1 = 0 \tag{4.27}$$
$$a_2 = 0 \tag{4.28}$$
$$a_3 = (x_e - x_s) \cdot \frac{10}{t_e^3} \tag{4.29}$$
$$a_3 = (x_e - x_s) \cdot \frac{-15}{t_e^4} \tag{4.30}$$
$$a_5 = (x_e - x_s) \cdot \frac{6}{t_e^5} \tag{4.31}$$

Figure 4.9 shows the desired trajectory, which is given to the controlled system in the simulation. It consists of several small trajectories and is selected in a way, that multiple different slopes are required and so that these alternate with times where the position is commanded to be hold. The functions for desired position and velocity can be transformed to Laplace domain and plugged into equation 4.24. By transforming the result back to the time domain, one obtains the theoretical curves for the position of the controlled system when using a ideal actuator with a continuous PD-control.

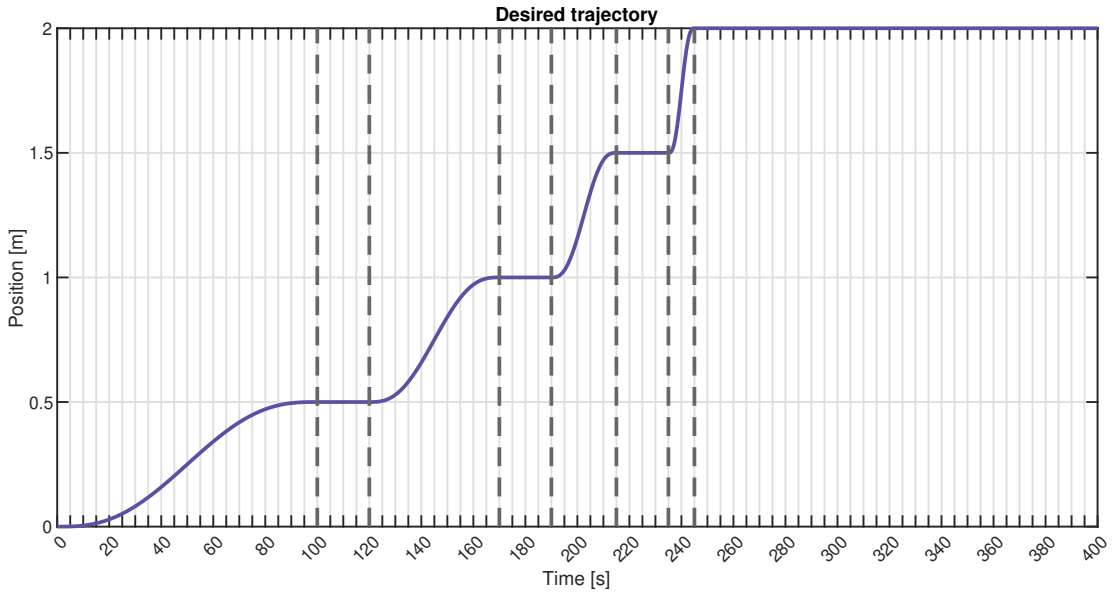


Figure 4.9: Progress of the position $x_{des}(t)$ of the used trajectory

Continuous Control Performance and Contouring Error For the closed loop system, Figure 4.10 visualizes the control behavior for both $x(t)$ and $v(t)$ of an exemplary trajectory.

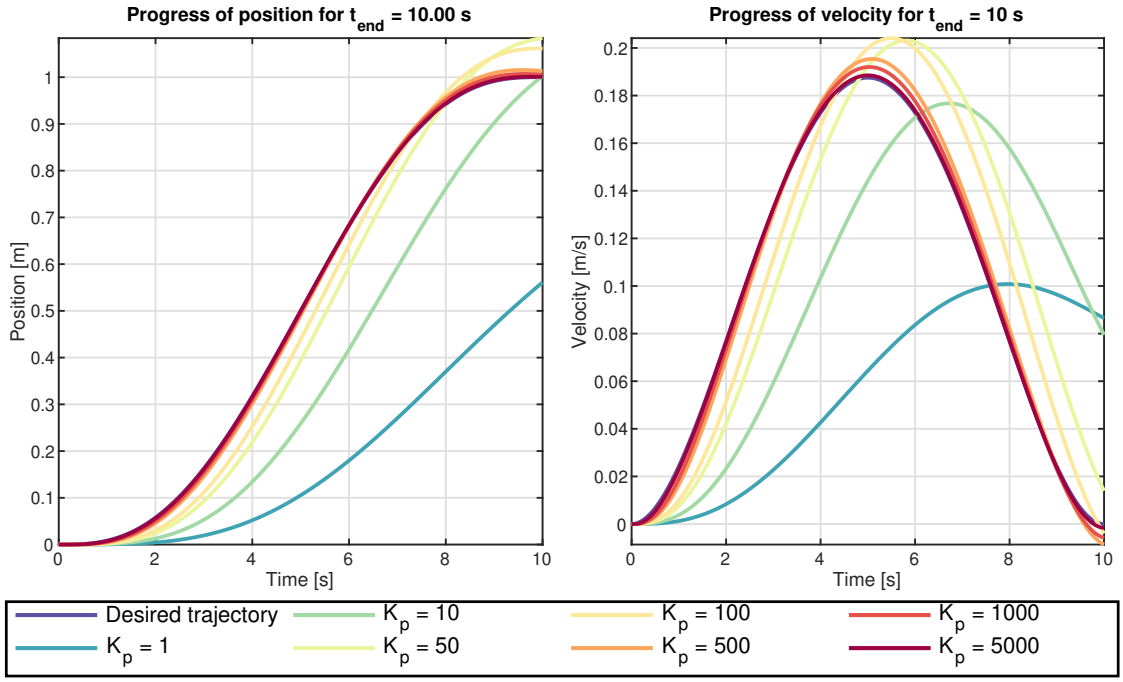


Figure 4.10: PD-control: Contouring error for a trajectory with $x_s = 0m$, $x_e = 1m$ and $t_e = 10s$ for different proportional gains

We can see the so called contouring error, identified by $x_{err}(t) = x_{des}(t) - x(t)$, and how it is reduced for larger selections of K_p. The controller requires an error in order to generate a force. For the same resulting force, the error needs to be lower for a higher stiffness of the virtual spring. Position and velocity always lack behind compared to the course of the desired input.

When analyzing the behavior of the discretely PD-controlled and thruster-actuated mass, we will not be comparing the position progress against the desired trajectory, but with respect to this analytical solution of the continuous PD-control. Thus, we can cut out this inherent slope-error, which is not occurring due to the discrete nature of both control and thrusters.

Additionally, we want to ensure that no saturation effects of the thruster actuator play a role in the analysis. Thus, at no point an acceleration that exceeds the one providable by the actuation should be required. Thus, the end time for each sub-trajectory must satisfy

$$t_e \geq \sqrt{\frac{10 \cdot \sqrt{3} \cdot (x_e - x_s) \cdot m}{3 \cdot u_{max}}} \tag{4.32}$$

This condition can be obtained by identifying the point of the maximal acceleration in $a_{des}(t)$ and using the knowledge of maximal acceleration of the used thruster configuration $a_{thr} = \frac{u_{max}}{m}$.

4.2.2 Boundary for Discrete Control

Analogously to the D-control example, a limit on the discrete control frequency can be specified by using the closed-loop time constant. The critically damped spring-damper-mass system has only one time constant

$$\tau_m = \sqrt{\frac{m}{K_p}} \tag{4.33}$$

In a similar fashion to before, this time constant is used in order to identify the required control frequency or the gain depending on the pre-specified parameter. For example, for a defined control frequency we can derive

$$K_p = \left(\frac{f_{ctrl}}{c_{samp}}\right)^2 \cdot m \tag{4.34}$$

using the coefficient c_{samp}, which specifies how high the control frequency is compared to the closed-loop time constant.

This concept was introduced for the D-control but can also be applied here. We recall that in general it is recommended to have a ten times faster control time interval than the fastest time constant in order to achieve a quasi-continuous control behavior for a discrete controller.

4.2.3 Time Discrete PD-Control with Thrusters

Similarly to the D-control, the performance of the PD-control has several limitations when operating with thrusters, as the usage of discretely controlled thrusters makes it nearly impossible to bring the velocity towards zero.

We will investigate the resulting dynamic behavior for two main operations, position holding and trajectory following. Under the presence of an initial velocity, for position holding a constant commutation within a bounded region is inevitable. A similar overview and analysis of the limit cycles occurring with PWM controlled thrusters is given in [71] with respect to attitude control.

Limit Cycles for Position Holding

After finishing the demanded trajectory, for most initial conditions and trajectories it is inevitable to continuously actuate in order to approximately hold the final position.

We remind ourselves that the simple thruster assignment logic leads to a discontinuity. Namely, assigned thrusts below u_{dead} occur and will be mapped to no actuation. In the PD-control, this means that a some point the system is not slowed down according to the continuous solution and exceeds the target position. After the virtual spring is loaded enough for the opposite thruster to minimally fire, we change the direction and a limit cycle can be observed.

Characterization of Different Phases The limit cycle is characterized by its amplitude, time interval and mean. While the amplitude and mean can be approximated by the actuation boundaries, the time interval of the limit cycle can additionally be deducted from the two velocities v_1 and v_2.

In the following, we will develop some intuition how the limit cycle can be shaped by selection of the open parameters. The goal is to identify equations that can give these four parameters dependent on the selected control gain K_p, f_{ctrl} and the fixed system parameters.

As depicted by Figure 4.11, one can identify four phases during the limit cycle:

1. Phase 1: The system is in free floating mode with a positive velocity v_1 and goes from the lower to upper actuation bound x_1.
2. Phase 2: At the upper bound, the system counter-actuates the dynamics and fires the thruster, which realizes a negative control force for a minimal time t_{mib}.
3. Phase 3: The system is in free floating mode with a negative velocity v_2 and goes from the upper to lower actuation bound x_2.
4. Phase 4: At the lower bound, the system counter-actuates the dynamics and fires the thruster, which realizes a positive control force for a minimal time t_{mib}.

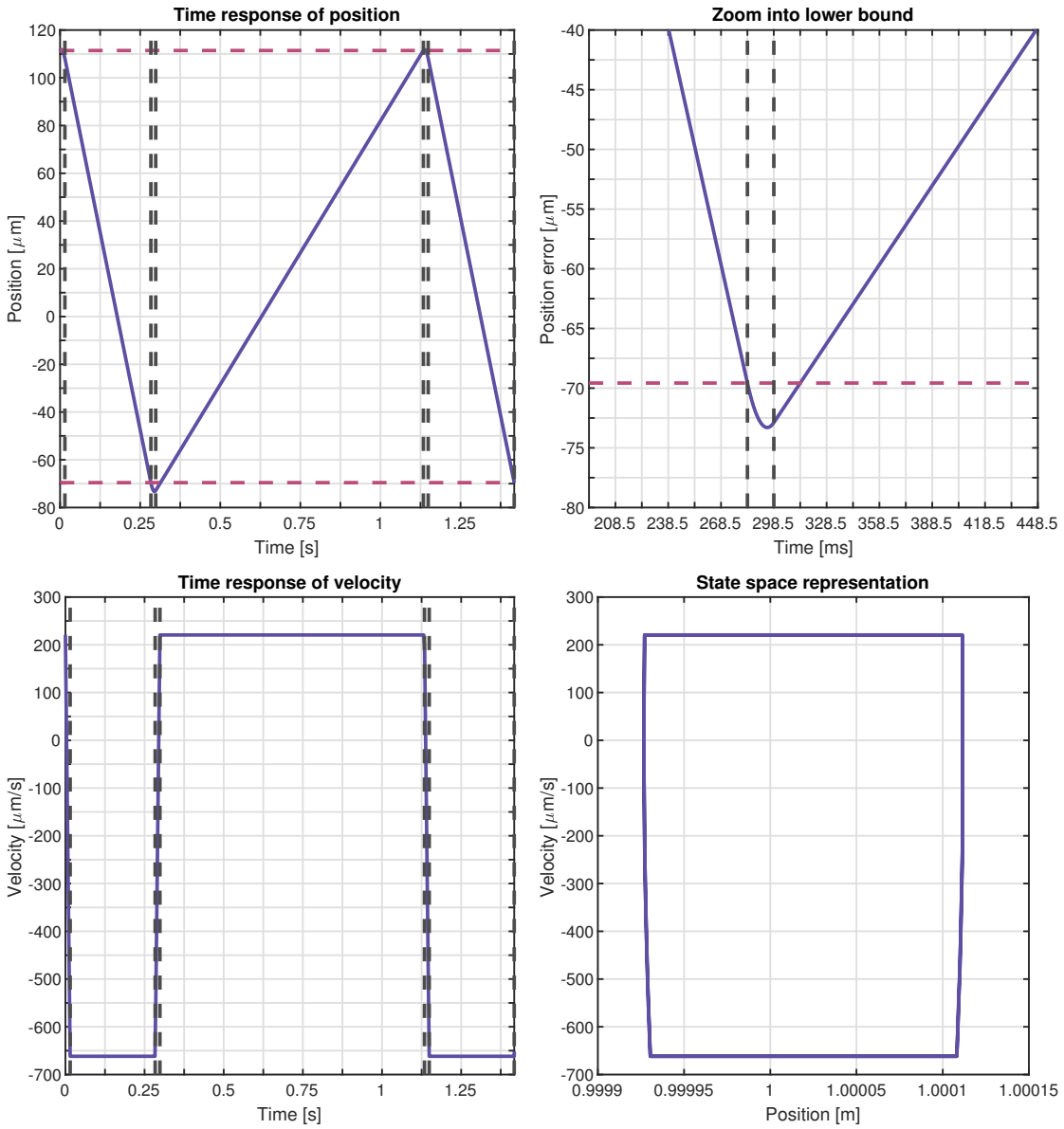


Figure 4.11: PD-control: Limit cycle behavior for $v_1 = 0.25 \cdot v_2$, $K_p = 75556$, $u_{dead} = 10N$ and $x_d = 1m$

The state space representation sums up the information of the position and velocity plot. The four phases are visible as four distinctive lines, which are repeated in each cycle. The switching is done at the four different corners. The results for the figure were obtained by commanding to constantly hold a position $x_d = 1m$ from a given starting position $x_s \neq x_d$ with an initial velocity being $v(0) = v_2 = 0.75 \cdot \Delta v_{mib}$. The K_p in the plot was set to such a large value in order to have a fast limit cycle for visibility of all phases.

Discrete Control with Perfect Firing First, we want to simplify the problem by assuming a discrete control with PWM thrusters which automatically triggers an ideal minimal firing when an assignment u_{dead} is reached. This means we cannot miss the minimal firing time point due to the control time intervals.

As noted before, the change in the velocity during the turning phases is a distinctive system parameter Δv_{mib} (see equation 4.9). Under the premise that the system immediately fires once u_{dead} is reached, one can parameterize v_1 and v_2 with respect to this system parameter.

$$\Delta v_{mib} = |v_1| + |v_2| \tag{4.35}$$

Thus, one can introduce the parameter d, with $0 < d < 1$, to represent v_1 by

$$|v_1| = d \cdot \Delta v_{mib} \tag{4.36}$$

and also v_2 by

$$|v_2| = (1 - d) \cdot \Delta v_{mib} \tag{4.37}$$

This kind of formulation assumes that the absolute velocities cannot exceed Δv_{mib} within the limit cycle. This intuition can be quickly verified with a simple example.

Let us imagine a scenario where the initial position is $x(0) = 0m$ and $v(0) = 0m/s$. At time $t = 0s$ a position of $x_d = 1m$ is commanded. Without an actuator with dead-zone, the critically damped system converges asymptotically to the goal position. The controller will first generate a positive force, which accelerates the system. At the turning point, a negative force is generated in order to slow down the system to asymptotically reach the goal.

With a dead-zone, the system will stop accelerating when the generated positive force drops below the dead-zone and only starts to decelerate for the first time when the absolute value of the negative control force is above the dead-zone. This will happen if we move closer to our desired position without reducing the velocity. The system minimally actuates the thrusters and will thus change the velocity by Δv_{mib}.

The last minimal actuation occurs within the limit cycle when the sign of the velocity finally changes, which requires $|v_1|$ to be smaller than Δv_{mib}. The exact value of v_1 is dependent on the previously commanded trajectory and the selection of the gains. Note that this requires a continuous control, as this does not necessarily apply if we miss the minimal firing time point. Therefore, the additional condition of not missing this point is needed for the discrete control.

Firing Conditions in Limit Cycle We remind ourselves that the control force of the critically damped PD-controller with commanded position holding is generated according to

$$F_{ctrl} = K_p \cdot (x_d - x) + K_d \cdot (v_d - v) = K_p \cdot (x_d - x(t)) - 2 \cdot \sqrt{K_p \cdot m} \cdot v(t) \tag{4.38}$$

Let us also recall that the limit cycle solely occurs due to the dead-zone on the thrust side u_{dead}. Due to the simplified mapping, this can be also directly expressed as a condition on the control forces. We can specify that a given control force is only actuated if

$$u_{dead} \leq |F_{ctrl}| \tag{4.39}$$

The firing condition for the upper bound x_1 is only achieved for positive velocities v_1. We will only reverse if the controller gives a negative force, which exceeds

$$-u_{dead} \geq K_p \cdot \Delta x_1 - 2\sqrt{K_p \cdot m} \cdot v_1 \tag{4.40}$$

with $\Delta x_1 = x_d - x_1$.

Reformulated with respect to the position differences, we can write

$$\frac{u_{dead}}{K_p} - 2 \cdot \sqrt{\frac{m}{K_p}} \cdot v_1 \leq -\Delta x_1 \tag{4.41}$$

Reversely, for negative velocities v_2, the system will only reverse at the lower actuation bound x_2 if the dead-zone is exceeded for the thruster, which realizes the positive control forces.

$$u_{dead} \leq K_p \cdot \Delta x_2 - 2\sqrt{K_p \cdot m} \cdot v_2$$

Reformulated, this gives

$$\frac{u_{dead}}{K_p} - 2 \cdot \sqrt{\frac{m}{K_p}} \cdot |v_2| \leq \Delta x_2 \tag{4.42}$$

by also using that v_2 is always negative.

Characterizing the Limit Cycle We can use the results for Δx_1 and Δx_2 to characterize the limit cycle. The limit cycle is ideally described with the position interval $x_1 - x_2$, the time interval and its average position, when we also know Δv_{mib},.

The position interval of the occurring limit cycle can be approximately given by ignoring the turning phases, where an additional position divergence takes place.

$$\begin{aligned}
-\Delta x_1 + \Delta x_2 = x_1 - x_2 &= \frac{u_{dead}}{K_p} - 2 \cdot \sqrt{\frac{m}{K_p}} \cdot |v_2| + \frac{u_{dead}}{K_p} - 2 \cdot \sqrt{\frac{m}{K_p}} \cdot |v_1| \\
&= 2 \cdot \frac{u_{dead}}{K_p} - 2 \cdot \sqrt{\frac{m}{K_p}} \cdot (1 - d) \cdot \Delta v_{mib} - 2 \cdot \sqrt{\frac{m}{K_p}} \cdot d \cdot \Delta v_{mib} \\
&= 2 \cdot \frac{u_{dead}}{K_p} - 2 \cdot \sqrt{\frac{m}{K_p}} \cdot \Delta v_{mib} \\
&= 2 \cdot \frac{u_{dead}}{K_p} - 2 \cdot \sqrt{\frac{m}{K_p}} \cdot \frac{u_{max}}{m} \cdot t_{mib}
\end{aligned} \tag{4.43}$$

The time interval can be approximated with high accuracy by

$$t_{cycle} = \frac{x_1 - x_2}{v_1} + \frac{x_1 - x_2}{v_2} + 2 \cdot t_{mib}$$

One can consult the previous results to get the time interval as a parameter of d and K_p.

$$t_{cycle} = \left(2 \cdot \frac{u_{dead}}{K_p} - 2 \cdot \sqrt{\frac{m}{K_p}} \cdot \Delta v_{mib}\right) \cdot \frac{1}{\Delta v_{mib}(d - d^2)} + 2 \cdot t_{mib} \tag{4.44}$$

Due to the symmetric acceleration phases, this approximation is accurate for $d = 0.5$. For other values of d, x_1 and x_2 are not the real positions where the system is located after the firing phase at the start of the free floating phase. Generally, this influence is very small though, as the acceleration phases are only a small time fraction and can also be ignored completely.

For $d \neq 0.5$, the average position of the limit cycle, will not be x_d. The average is shifted towards the bound with smaller absolute velocity value and can be approximated directly by calculating x_1 and x_2. From the equations 4.41 and 4.42 one can immediately see that the x_d will not necessarily be located inside the actuation bound.

Tuning of the Proportional Gain We can use equation 4.43 in order to specify the tolerated actuation bound in the limit cycle. For a selected control frequency the proportional gain can be calculated accordingly.

The tightest actuation bound can be derived if we set $x_1 - x_2 = 0$ and solve for K_p. This results in

$$K_{p,max} = f_{ctrl}^2 \cdot m \tag{4.45}$$

which equals a value of $c_{samp} = 1$. By selecting this value, no free floating phases can be observed anymore but a constant alternating actuation of the two thrusters.

By assuming the starting conditions of a system, which is commanded to hold a position, are set to $x_d = x$ and $v_d = 0m/s$, the proportional force can be dropped and one gets

$$u_{dead} \leq 2 \cdot \sqrt{K_p \cdot m} \cdot |-v| \tag{4.46}$$

Stated as a velocity bound

$$|-v| \geq \frac{u_{max} \cdot t_{mib} \cdot f_{ctrl}}{2 \cdot \sqrt{K_p \cdot m}}$$

It is sensible to limit the proportional gain, so that the controller does not immediately fire for velocities below Δv_{mib}. The system could otherwise overact already at the start. The limit for K_p is thus given for

$$Kp_{max,vonly} \leq \frac{m \cdot f_{ctrl}^2}{4} \tag{4.47}$$

and was obtained by plugging equation 4.9 into the velocity bound and solving for K_p. This inequality will be also important for the general case later on. This condition would require $c_{samp} = 2$.

Examples for Location and Shape of Limit Cycle In the following, some exemplary plots for different possible limit cycles shall be given.

Note that these are only valid for the discrete control with PWM actuated thrusters under the premise of perfect firing on the dead-zone force. The proportional gain was selected according to equation 4.43 for a given control frequency and bound $x_1 - x_2$.

Figure 4.12 visualizes how the dead-zone is shaped for a bound $x_1 - x_2 = 1mm$. On the left side, we can see what happens if $d = 0.1$, while the right side depicts $d = 0.5$. One cannot give a guarantee for which value d we converge to the dead-zone, as this is dependent on the previous operations. Thus, this value can be seen as a randomly drawn variable $0 \leq d \leq 1$. We can see that the limit cycle time interval heavily relies on this value. The fastest limit cycle is observed for $d = 0.5$, which can be also verified by equation 4.44. In this case, independently from the control frequency, the limit cycle is also symmetric and centered around x_d.

For $d \neq 0.5$, the limit cycle is centered around another value and x_d is not necessarily included within this interval. In addition we can observe differences for different control frequencies, which comes from the different resulting values for u_{dead}.

Figure 4.13 shows that for the tightest possible upper and lower bounds, a continuous firing is required. No free-floating phases can be observed. Once again, the centering of the limit cycle is dependent on f_{ctrl}.

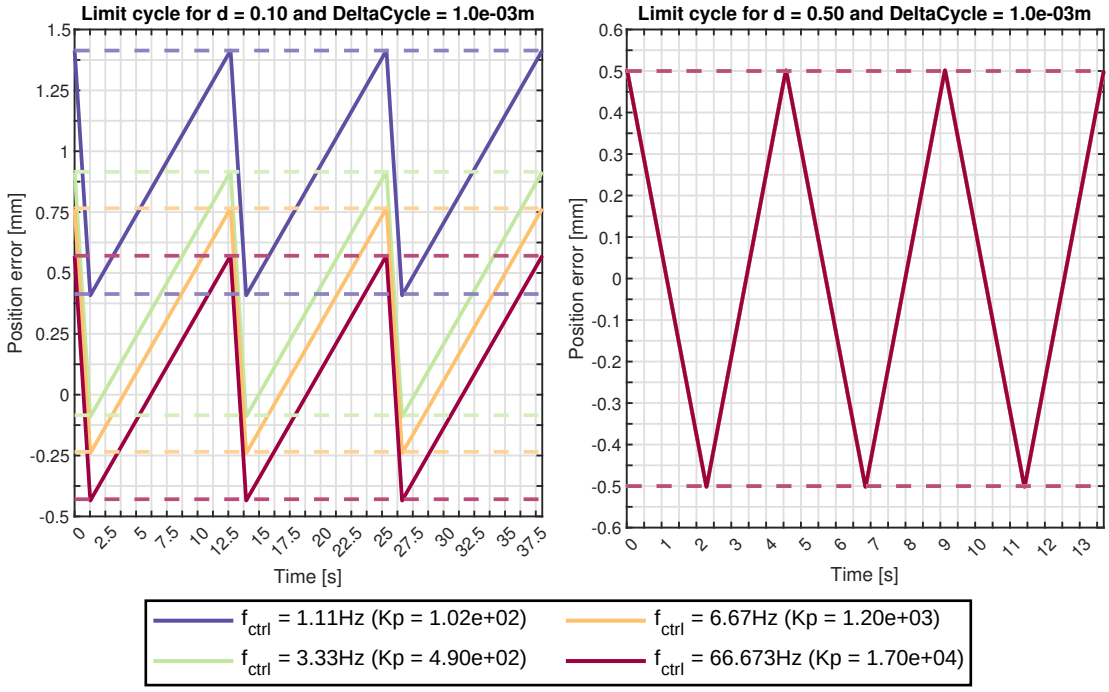


Figure 4.12: PD-control: Limit cycles for $v_1 = d \cdot \Delta v_{mib}$ and $x_d = 1m$ (K_p follows from equation 4.43)

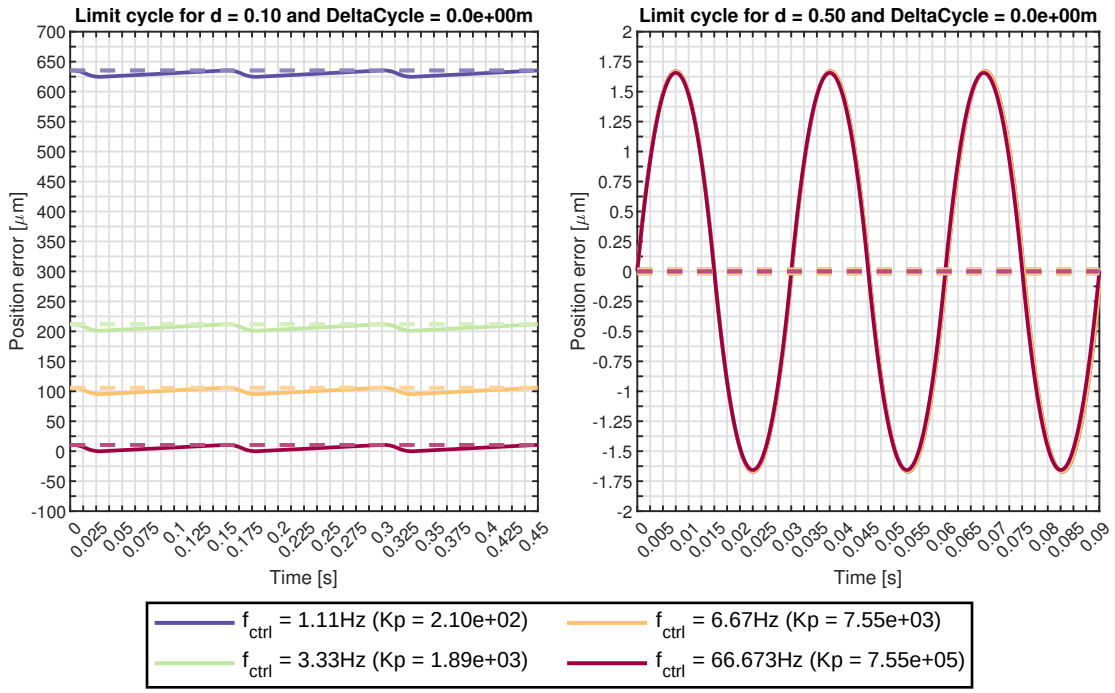


Figure 4.13: PD-control: Limit cycles for tightest bound and $v_1 = d \cdot \Delta v_{mib}$ and $x_d = 1m$ (K_p follows from equation 4.43)

Limit Cycles in General Discrete Controlled Systems For real applications, it cannot be guaranteed that the system actuates perfectly at u_{dead} when coming from a free-floating phase. It is possible that the optimal point in time, at which the assigned thrust exceeds the natural dead-zone, is missed. In such a case, the resulting applied control force will potentially exceed u_{dead}. Then, the change in velocity is not exactly Δv_{mib}, but a higher value.

As a result, not only the time interval of the limit cycle is reduced, but also the actuation bounds are changed. The absolute values for v_1 and v_2 do not necessarily sum up to Δv_{mib} anymore and the limit cycle is not fixed. This case can be seen as over-firing. Due to the longer control time intervals, this problem becomes worse for low control frequencies.

Let us assume the worst-case. The firing was only infinitesimally missed in one control time interval k with, for example, $u_1(k) \approx u_{dead}$. The control force gets larger due to the on-going free floating period with an unknown velocity v_{free}. In the next control time interval an increased control force is generated and thus the desired thrust increases

$$u_{new}(k+1) = u_{dead} + |K_p \cdot \frac{v_{free}}{f_{ctrl}}| \tag{4.48}$$

An over-firing takes place if this thrust is not mapped to u_{dead} after the discretization done by the PWM. Now it is harder to make definitive statements regarding convergence and limit cycles.

The influence however is only marginal, as long as the proportional gain does not exceed the one specified in equation 4.47. The limit cycle can still be seen as approximately constant, as it only jitters marginally at each control time interval.

So overall, the previously deduced intuitions of the parameter tuning can still be used. Gains closer to the maximum of $K_{p,max} = f_{ctrl}^2 \cdot m$ are more critical in general.

One can also use equation 4.48 to tune f_{pwm} with respect to the limit cycle behavior. By plugging Δv_{mib} into v_{free}, one gets sensible results for a good maximal discretization which should be aimed for.

Figure 4.14 shows some selected results for the conducted simulation by investigating only the simulation time, starting from $t = 270s$. This means only the last position holding time interval is plotted.

The state space specifies the limit cycle. We can see that some cycle times are very long, as for some setups not even one limit cycle was finished during the 130s of remaining simulation time. We can also see that the time intervals are not absolutely fixed for discrete control but change slightly after every control time interval.

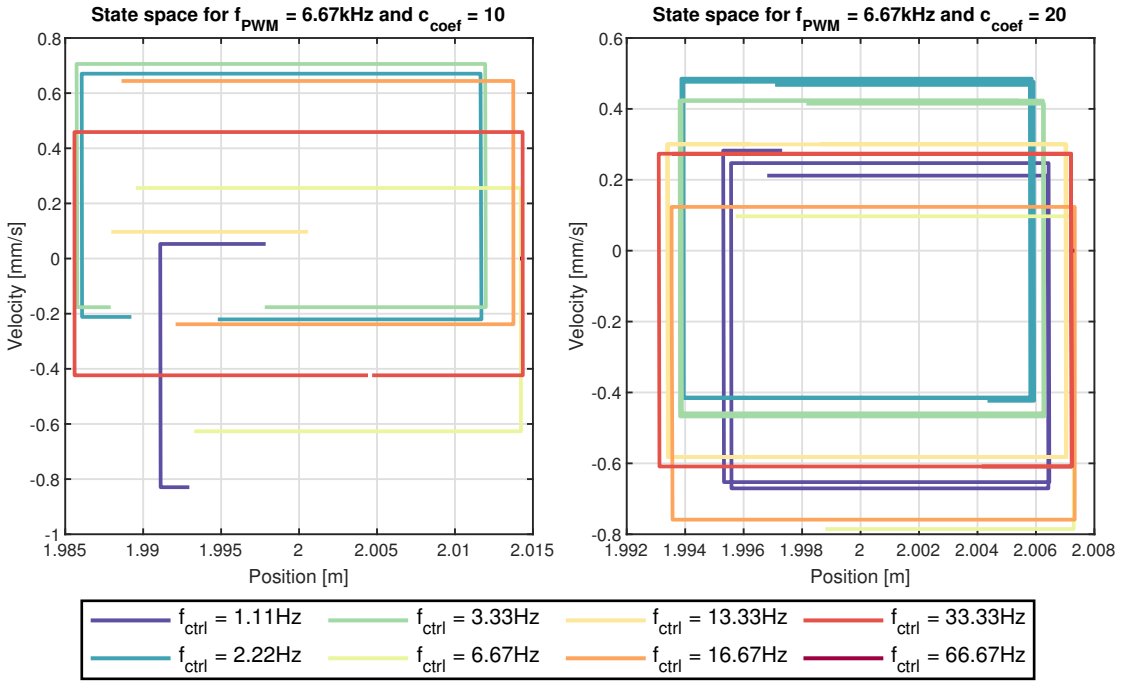


Figure 4.14: PD-control: Limit cycle behavior from simulation starting at t=270s with $K_p = c_{coef} \cdot f_{ctrl}$

Trajectory Following

The performance of the discrete PD-control with PWM thrusters is very similar to the continuous one with ideal actuators. The error is pre-dominated by the contouring error, which occurs for all PD-controls without a feedforwarding of the dynamics, often referred to as "+"-part.

The proportional gain can be enlarged in order to reduce this error, as the virtual spring gets more stiff and then tolerates only smaller errors. This can be also seen for the two exemplary setups in Figure 4.15. The left side has a ten times lower gain than the right side, which is specified by $K_p = c_{coef} \cdot f_{ctrl}$. Figure 4.16 directly depicts the position error $x_{des}(t) - x(t)$.

We can observe the typical contouring error of a PD-controller for this type of trajectory. First, the position lags behind. After the turning point, the positions run ahead. Every rise in the desired trajectory can be identified this way. By increasing c_{coef}, the error is successfully reduced.

A steeper slope in the desired trajectory increases the control error more heavily. The actuation is too slow, especially for smaller control frequencies, which explains the large errors for the fast trajectories in the end.

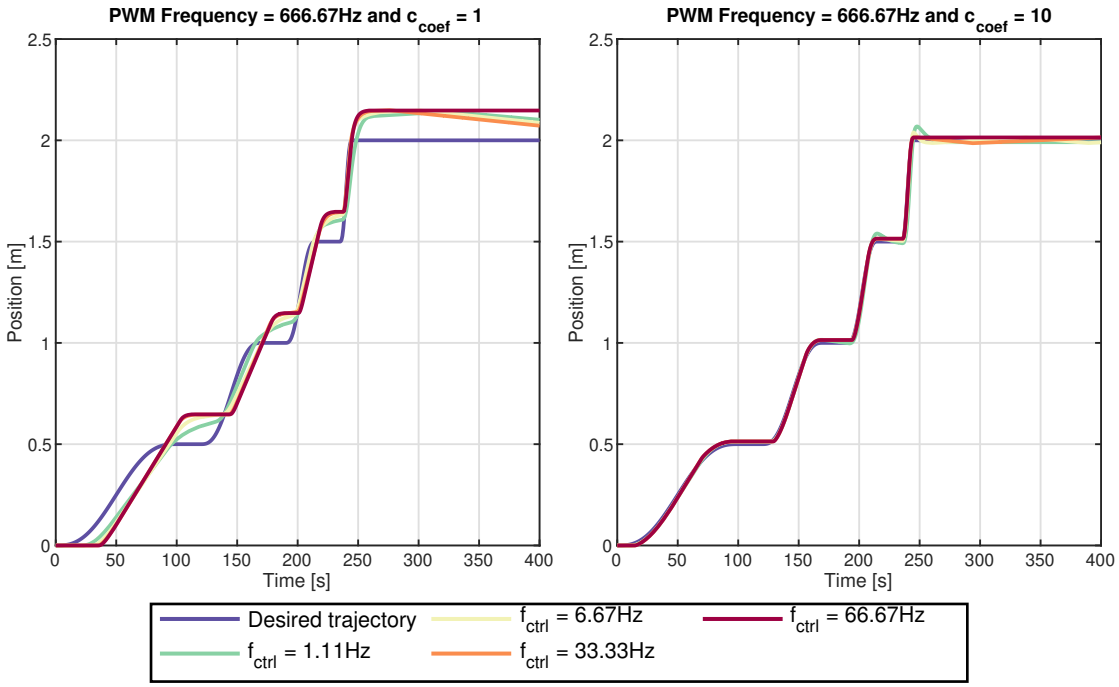


Figure 4.15: PD-control: Time response of position for trajectory following (with $K_p = c_{coef} \cdot f_{ctrl}$)

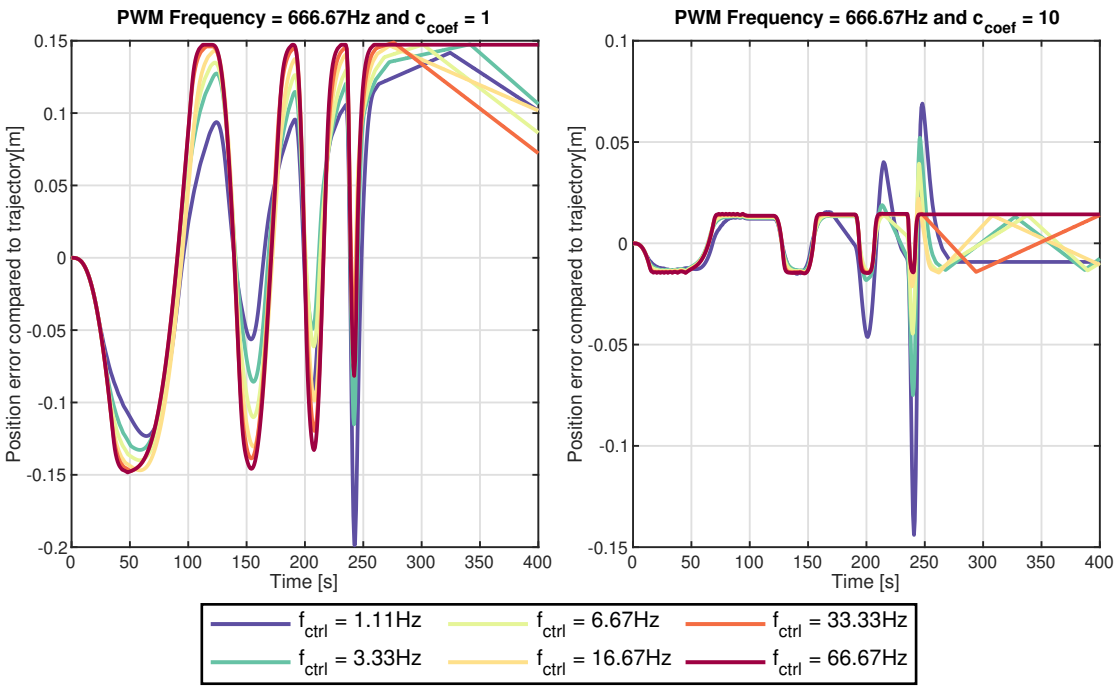


Figure 4.16: PD-control: Time response of position error for trajectory following (with $K_p = c_{coef} \cdot f_{ctrl}$)

For $f_{ctrl} = 67Hz$ no limit cycle is observed, as the control time interval equals exactly the minimal firing time. Thus, no discretization of the pulse lengths within the PWM is done. All acceleration due to firing of the thrusters can be automatically inverted. Without an initial velocity, the system can therefore stop. This will always happen when the timings of the pulses of the controller are equal to multiples of t_{mib} (e.g. no limit cycle for all used control frequencies for $f_{PWM} = 67Hz$).

As a partial result, one can also observe how the limit cycle is affected by rising K_p. The tighter bound forces a shorter time interval.

Fuel consumption

The required acceleration for a trajectory gives us a good indication how much fuel the optimal force actuator would require. By comparison with the real firing times of the thruster actuated system, we can see that fuel is mainly wasted during the actuation of the limit cycles.

This also stresses the importance in handling and shaping the limit cycles. A constant oscillation between lower and upper limit of the thruster position is not desired. A larger time interval of the limit cycle reduces the number of required corrections and thus minimizes the fuel consumption.

4.3 D- and PD-Control with Pulse-Width Pulse-Frequency Thruster

The intuitive parametrization of the PWPF modulator, with r_{min} being the only open parameter, allows a usage very similar to the PWM modulator. r_{min} can just be applied in a similar fashion as u_{dead}. The same simulation setups which were previously described for the PWM modulator can be carried out identically with a PWPF modulator.

4.3.1 D-Control

In the PWPF modulator there is no relation between the control frequency and r_{min}. This means that we just have to set f_{ctrl} sufficiently high with respect to the closed-loop time constants. We select

$$f_{ctrl} = \frac{K_d}{m} \cdot 10 \tag{4.49}$$

We still have to set K_d and r_{min}. For the simple thruster assignment, we can give a physical relation using the minimal change in velocity. We should only actuate if the remaining velocity is above the minimal change in velocity.

$$u_{max} \cdot r_{min} = c_{fac} \cdot K_d \cdot \Delta v_{mib} \tag{4.50}$$

with c_{fac} being a modulation factor, which tunes the final velocity.

It can be easily shown that this parameter has to fulfill $c_{fac} \geq 1$. Otherwise a smaller velocity than Δv_{mib} triggers a pulse that leads to a change in the systems velocity equal to Δv_{mib}. This means the velocity can go below zero and the system does not converge. A limit cycle can be observed for the D-control, which is undesired.

For $c_{fac} \geq 1$, the control performance is comparable to the PWM controlled thruster simulation and was tuned by an equivalent coefficient according to equation 4.8. The main difference lies in the convergence rate, as the time intervals of the thruster pulses grow towards infinity for assignments close to r_{min} in the PWPF modulator. This leads to a general slower reduction of the velocity. The lowest final velocity can be reached for $c_{fac} = 1$ and results, as before, in the boundaries given by Δv_{mib}.

For the PWM thrusters it was possible to further reduce the final velocity bound by using a more advanced thruster assignment which modeled u_{dead}. This adaption however cannot be done for the PWPF modulator, as the pulse time intervals of the different thrusters are not synchronized.

If we use such an assignment method, we can, without loss of generality, imagine that the pulse frequency of the positive thruster is higher than the negative orientated thruster. It is possible that one thruster fires during one control time interval, but the other one does not. This significantly changes the assignment for the next control time interval and means that we cannot carry out small relative pulse lengths due to the missing synchronization.

For 6 DoF wrenches, one can imagine that this problem becomes even more present due to the potential coupling between the dimensions. The tuning of the gains becomes more delicate, as it has to be prevented that the different thrusters continuously fight each other.

4.3.2 PD-Control

The same reasoning can be applied for the PPWF thrusters and PD-control. Again, the analogy between

$$r_{min} \cdot u_{max} \triangleq u_{dead} = f_{ctrl} \cdot u_{max} \cdot t_{mib} \tag{4.51}$$

is used. A sufficiently high control frequency is used in comparison to the closed-loop time constant. The condition 4.47 is translated to

$$K_p \leq \frac{r_{min}^2 \cdot m}{4 \cdot t_{mib}^2} \tag{4.52}$$

The PWPF will give similar outputs as the PWM, also for PD-control, as long as this limit is satisfied; even though analytical analysis is here more difficult.

Analytical conditions on the limit cycle are difficult to give. The exact firing point is not easily determinable, as it depends on the previous internal state of the output of the pre-filter, the change of the input of the modulator $r(t)$ and the defined r_{min}. It is not guaranteed that the thruster fires during one control time interval so that the control force will enlarge until the firing is conducted. Potentially, the PWPF modulator is thus more prone to overfiring, depending on the proportional gain.

Practically, one can observe similar limit cycles for controlled position holding. They are approximately specified by Δv_{mib} and change slightly over time due to the discrete nature of the control. The assigned thrust soon rises towards inputs above r_{min}, which are realized with pulse frequencies in the area of f_{ctrl}, so that no significant loss in performance is observed.

The same can be said for trajectory following mode. In the end, the PWPF modulator mainly introduces a small time delay, which comes from the long pulse intervals close to r_{min} and is quickly overcome in dynamic usage.

4.4 Relevance to Multi-Dimensional Settings

The previous descriptions laid out how the severity of multiple limitations of the thruster-controlled systems like limit cycles or non-zero final velocities can be reduced for the one-dimensional example.

Usage of more powerful assignment methods, which better consider the thruster properties, can effectively lead to an employment of the full controllability envelope of the actuation system. We saw for the very simple two-thruster configuration how the fuel consumption is not distributed as nicely as for an ideal force actuator.

In fact, we have forces that are preferred by the actuator and require less fuel. This problem becomes more prominent for the three-dimensional case, where we have a coupling between the different dimensions.

For the PD- and D-control, we developed relations which allowed us to tune the open parameters of the system in accordance to the desired performance, while in parallel reducing the fuel consumption. Practically, we forced the controller to demand better control forces with respect to the system dynamics.

Properties like the minimal change in velocity are more difficult to be formulated in a multi-dimensional setting. Also the occurring limit cycles will be not as regular as the observed ones. This is due to the more complex mappings from wrench to thrust space in this setting. However, by identification of the minimally applied thrust in each dimension and depending how the different thrusters are geometrically distributed, similar properties could be derived for the 3D case.

In future works, a Passivity Observer and Passivity Controller (POPC) approach may be used to address the multidimensional case by managing the energy content in the limit cycle injected by the non-linearity of the thruster deadzone.

5

Application in Robot Simulator

The space robot operates in a a three-dimensional environment, which means that the previous one-dimensional point of view has to be extended to six degrees of freedom motions. As mentioned in the introductory chapter, the combination of a base spacecraft together with a robot arm results in a complex and coupled structure. The external momentum, induced by momentum transfer through contact with other objects, leads to a drift of the robot and, in order to be counteracted, requires the use of thrusters.

In the following, we will use a newly developed controller that minimizes the consumed fuel, while still keeping the workspace fixed. For this controller, the performance for an exemplary control task is simulated in a realistic environment, which is introduced in more detail in Section 5.2.

The knowledge of how to optimally select and actuate thrusters will be utilized in order to reduce errors coming from the thruster characteristics. The redundant configuration of 24 thrusters, which has a minimal coupling in the wrench parts, was selected. This configuration has already been analyzed statically with respect to different point of views throughout the previous sections (see e.g. Subsection 2.5.2). For determination of the thruster assignment, the newly developed Mixed Integer Programming method will be used. Looking forward, we will be able to validate the functionality of this method for a realistic environment.

A more detailed description of the simulation parameters will be given in Section 5.3, where we will investigate how the thruster actuation affects the control performance. We will observe a more complex extension of the one-dimensional limit cycles.

5.1 Momentum-Based Control

In the introduction chapter 1, it was already mentioned that the standard control approaches of space robots have either the problem of excessive fuel consumption, when using a control-scheme similar to the free-flying manipulator, or the problem of leaving the workspace due to an unchecked drift, as present in free-floating schemes.

The space robot starts drifting due to transfer of momentum during a contact with the target object or due to a remaining translational or rotational velocity after approaching the target.

As a counter measure, momentum dumping was introduced in [72]. The build-up momentum is extracted, while the end-effector can be controlled simultaneously. This approach is based on a decomposition of motion into internal and external motions. The thrusters are only used to dump momentum and not to counteract internal forces stemming from the manipulator. Consequently, the drift can be stopped.

Though, this cannot be done immediately and the base comes to rest only after some time. After prolonged operation, the manipulability of the robot arm diminishes and we can still have the problem of reaching the boundary of the robot workspace.

Thus, this concept is extended in [73], where not only the momentum is dumped but also the workspace is stabilized. The COM location of the complete system is controlled around a desired inertial location. The manipulator is guaranteed to reach the target for all times. In the following, only the basic control scheme shall be introduced. For an extensive overview the interested reader is referred to [73] (see also [74]).

Shortly, we can introduce the general dynamics of the space robot by

$$\mathbf{M}(q) \begin{bmatrix} \dot{v}_b \\ \dot{\omega}_b \\ \ddot{q} \end{bmatrix} + \boldsymbol{C}(\boldsymbol{q}, v_b, \omega_b, \dot{\boldsymbol{q}}) \begin{bmatrix} v_b \\ \omega_b \\ \dot{q} \end{bmatrix} = \begin{bmatrix} f_b \\ \tau_b \\ \tau_m \end{bmatrix} \tag{5.1}$$

We can see that we can partition the dynamics into base and manipulator contributions, which are coupled. The linear and angular velocity of the base is given by v_b and ω_b, respectively, in the body frame. Both parameters are vectors defined in $\mathbb{R}^3$. Consequently the wrench of the base can be given by its division in force and torque parts $f_b, \tau_b \in \mathbb{R}^3$. Analogously, as our robot arm has seven joints, the manipulator is defined by its joint angles $\boldsymbol{q} \in \mathbb{T}^7$ and joint velocities $\dot{\boldsymbol{q}} \in \mathbb{R}^7$. $\mathbb{T}^7$ is the seven-dimensional torus consisting of all feasible joint angles.

Note that the inertia matrix $\mathbf{M} \in \mathbb{R}^{13x13}$ consists of various sub-matrices. We can discriminate between the matrices on the diagonal, which correspond to the uncoupled inertia matrices of the base and manipulator, and also the coupling inertia matrices. A similar identification can be conducted for the Coriolis and centrifugal matrix $\boldsymbol{C}$. At this point we forgo a further consolidation of the exact formulation of the matrix entries. Once again, be referred to [73] or similar literature. Here, it is assumed that the manipulator will always have seven joints, but this consideration is also valid for a general number of n joints.

In the current state, the control inputs are equal to the right side of equation 5.1. By deriving a triangular relation, this allows the introduction of the desired control inputs. The system is decomposed into an external part, being the complete floating system, and the internal motion of the end-effector.

Put shortly, we obtain

$$\begin{bmatrix} f_b \\ \tau_b \\ \tau_m \end{bmatrix} = \begin{bmatrix} R_{cb}^T & 0 & 0 \\ \hat{p}_{cb} R_{cb}^T & R_{cb}^T & 0 \\ \bar{J}_v^T R_{cb}^T & \bar{J}_\omega^T R_{cb}^T & J_m^{*T} \end{bmatrix} \begin{bmatrix} f_c \\ \tau_c \\ \omega_{e,int} \end{bmatrix} \tag{5.2}$$

The new control inputs are given by the centroidal force $f_c \in \mathbb{R}^3$, the angular acceleration of the complete system $a_r = I_c^{-1} \tau_c \in \mathbb{R}^3$ and the internal wrench of the end-effector $\omega_{e,int} \in \mathbb{R}^6$. J_m^{*T} is the generalized Jacobian of the end-effector. The hat operator symbolizes the generated skew-symmetric matrix, which can be given for any vector. R_{cb} is the rotation matrix between body frame and COM frame. The exact relations for $\bar{J}_\omega$, $\bar{J}_v$ and p_{cb} will not be given here, but can also be derived from calculation of the total momentum around the COM frame of the complete system.

The change in coordinates allows us to introduce some simplified control laws.

First, we remind ourselves that we wanted to directly control the end-effector. This is done by matching the end-effector frame with the frame of the desired pose. The internal end-effector wrench is generated according to a PD-like control law.

$$\omega_{e,int} = -J_{\tilde{x}_e v_e}^T K_e \tilde{x}_e - D_e v_e \tag{5.3}$$

Note that the rotation is represented by using the vector part ϵ of the quaternion representation. The translational and rotational error in the frames is given by $\tilde{x}_e$. The time derivative of this value can be obtained by multiplying v_e with the coordinate representation Jacobian $J_{\tilde{x}_e v_e}$, which can be given by

$$J_{\tilde{x}_e v_e} = \begin{bmatrix} I & 0 \\ 0 & \eta I - \hat{\epsilon} \end{bmatrix} \tag{5.4}$$

with η being the scalar part of the quaternion representation. As in every PD controller, the stiffness matrix K_e and the damping matrix D_e are positive definite matrices. In the case of a three-dimensional control they are consequently $\in \mathbb{R}^{6x6}$.

Second, the centroid force results from another PD-like control law

$$f_c = K_c \tilde{p}_e - D_c v_c \tag{5.5}$$

where $\tilde{p}_e$ is the difference between desired centroid position and the current one. Due to the reduced dimensionality K_c and D_c are only $\in \mathbb{R}^{3x3}$, though both are also positive definite.

Third, singularities can only be avoided if also the angular momentum is dumped. Thus, we generate the net torque around the centroid τ_c

$$\tau_c = -D_{hr} h_r \tag{5.6}$$

by using the angular momentum h_r and another positive definite damping matrix.

Combined, these control laws result in our desired control approach.

5.2 Robot Simulator and Simulation Setup

The experiments were conducted in a simulation environment which was developed as part of a hardware-in-the-loop simulation facility for OOS at the German Aerospace Center (DLR) [75]. Real space operations can be simulated with a high level of realism using overall three different stationary robots and mission-like computational hardware. Two KR120 robots are used to model client and servicer, while a Light Weight Robot (LWR) corresponds to the manipulator present in real space robots.

Due to the limited scope of this work, no simulations involving hardware-in-the-loop could be conducted. Here, we solely rely on the results of a Simulink model, which uses modeled values of the dynamics. As we will also remain on a qualitative point of view in the following consideration, it is not deemed necessary to introduce the complete simulation setup, but only a quick overview will be given.

The exemplary control task, which will be investigated, consists of moving the manipulator from an upwards position downwards to a gripping mode. Start and goal configuration are visualized in Figure 5.1. We can see that the base moved due to the initial velocity. The joints in the arm get adjusted accordingly to reach the goal pose in the end-effector.

Figure 5.1: Robot Simulator: Example realization of the task, Initial (left) and end configuration after 10s(right)

For this control task we will be using the previously introduced momentum-based controller, which is additionally extended by a nullspace optimization. Note that the current problem definition would mean that no usage of the thrusters is required, as the position of the COM of the complete system is commanded to be constant and no external contact is present. However, additionally to the end-effector task, we have an initial translational $\boldsymbol{v}_{b,i} = [2 \;\; 0 \;\; 2]\frac{mm}{s}$ and rotational $\boldsymbol{\omega}_{b,i} = [0 \;\; 2 \;\; 0]\frac{deg}{s}$ velocity present in the base. This means that a control response which stabilizes the system will be required.

As previously stated, we will use the highly redundant 24 thruster configuration, which consists of thrusters with $u_{max} = 10N$ and $t_{mib} = 15ms$. The control loop, which updates the thruster assignments, runs at a frequency of 3.33Hz.

The thrusters are assigned according to the Mixed Integer method (see MILP2 in 2.5.1) and modulated with a PWM with $f_{pwm} = 333.3Hz$. The mass of the base is equal to 300kg, while the different links of the manipulator together with the gripper weigh around 17kg as well.

As stated, we are mainly interested in the qualitative results, thus the exact values of the inertia of the different parts will not be given.

5.3 Simulation Results With Ideal Model

In short, our control task in this scenario consists of two sub-tasks. On the one hand the end-effector should traverse from initial to target pose and hold this pose after convergence. On the other hand, the initial drift has to be stopped so that the initial position of the system COM is kept. This requires a dumping of the accumulated momentum.

We are mainly interested in the question of how the control errors change over time. Thus, it is not necessary to add further details of, for example, the exact initial configuration.

We recall that the end-effector task is realized similar to an Impedance controller in Cartesian space. Figure 5.2 shows that the end-effector task can be solved successfully. Only a neglectable error can be observed after 20s of simulation time for both the position and the roll-pitch-yaw angles.

The position and orientation of the base is not directly controlled. The base is allowed to move freely but is indirectly affected through the control of the position of the COM of the centroid. Thus, in Figure 5.3 the current base parameters are referred to the initial base pose. We have an initial translational velocity along x- and z-axis in combination with a rotation around the principal y-axis at the base. This can also be seen in the figure by observation of the initial slopes.

During the control operations, the base state is also indirectly affected. We can see that the rate of change for both position and orientation stabilizes only after the end-effector converged after around five seconds. The ongoing change of the position and orientation error in the end indicates a drift of the base.

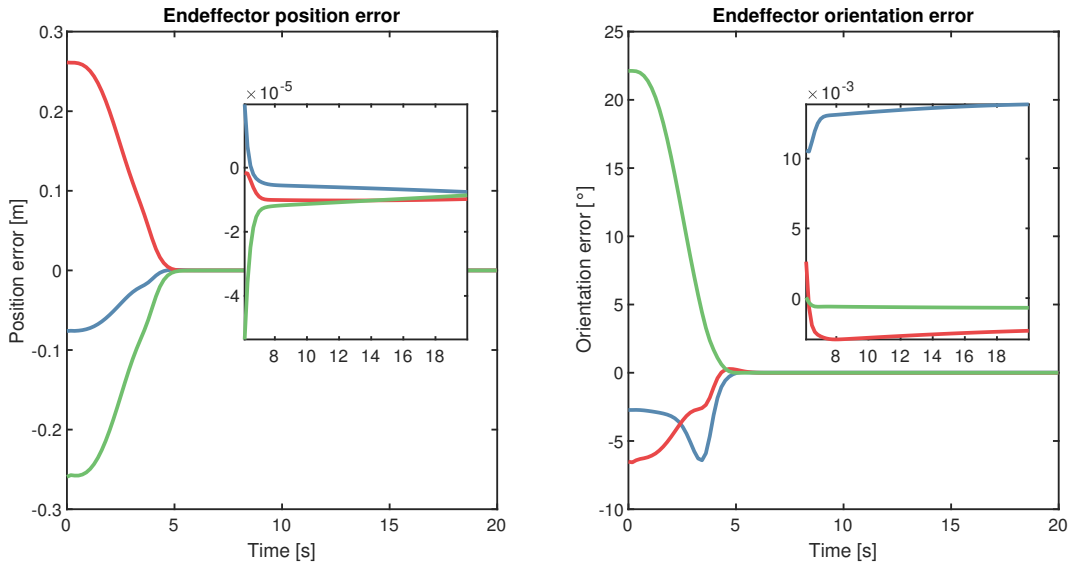


Figure 5.2: Robot Simulator: Position and orientation error in end-effector (blue = x-axis, red = y-axis and green = z-axis)

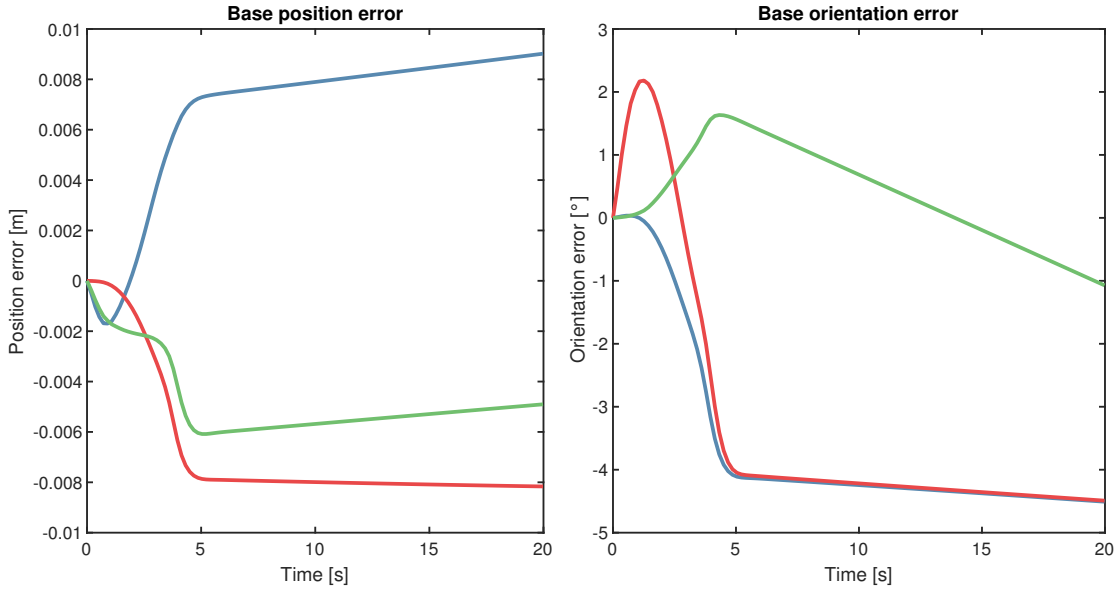


Figure 5.3: Robot Simulator: Position and orientation error in base (blue = x-axis, red = y-axis and green = z-axis)

Figure 5.4 depicts how well the secondary control task is fulfilled. We can see in the graph on the right side that the angular momentum is reduced significantly. Without any other actuator available, the angular momentum will be removed using the thrusters. This operation requires some fuel, which can be seen in Figure 5.5. This plot shows the consumed fuel, which is, as before, calculated from the integrated sum of thruster assignments in each timestep. However, the small accumulated fuel value shows that overall only a few thrusters were required for a small time frame.

The COM position, as visible in the left plot in Figure 5.4, is not perfectly stabilized but a small velocity remains in the system. We are reminded of the one-dimensional limit cycle from Section 4.2. The system oscillates between a lower and upper position bound due to the limited modulation resolution and the minimal firing time t_{mib}. When a control wrench threshold is exceeded, a thruster actuation is triggered. The observation time however is to small in this case. Later on we will see faster and more complex limit cycles when model uncertainties are introduced.

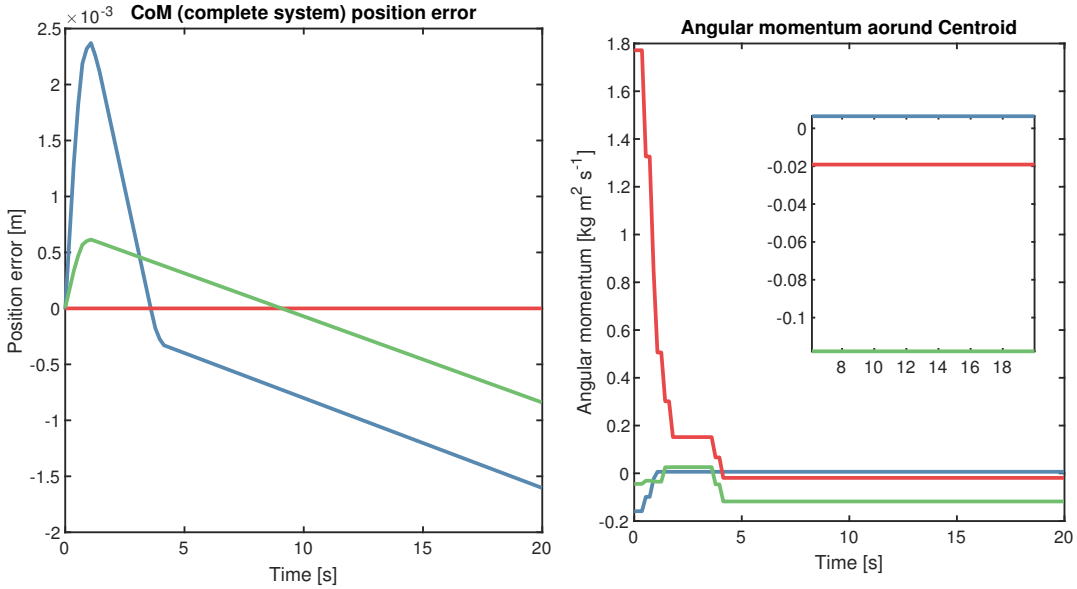


Figure 5.4: Robot Simulator: Position error and angular momentum of centroid (blue = x-axis, red = y-axis and green = z-axis)

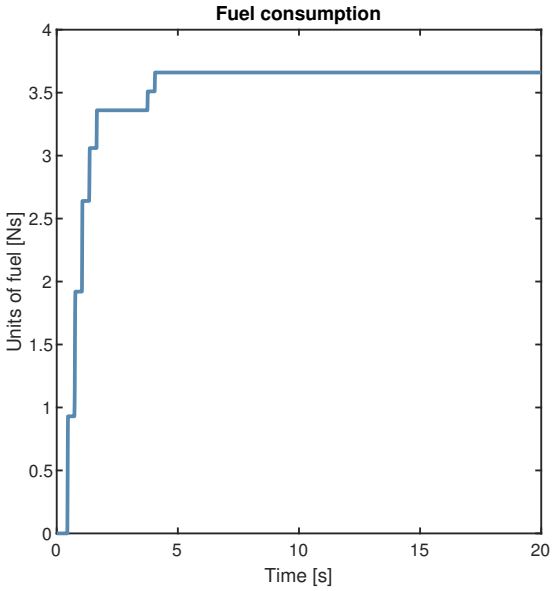


Figure 5.5: Robot Simulator: Accumulated fuel consumption during operation

5.4 Modeling of Uncertainties

In the previous section, it was always assumed that specified nominal parameter values are optimally realized in the real system. No sources of error were regarded.

In reality, uncertainties in the state measurement and deviations in the setup parameters diminish the quality of the control. The effort in guaranteeing a high performance due to choosing optimal thruster assignments might become futile under an erroneous system. Potentially, a disadvantageous combination of model errors leads to a violation of the mission requirements on the control side. In the worst case, the controlled system can even become unstable.

In the following, we will shortly investigate the most important sources of errors for the space robot. Under the premise of uncertainties, the complexity of the system makes it difficult to give a definitive statement regarding the control performance. It has to be verified that the spread of all possible final control behavior safely meets the project specifications. We will use a Monte-Carlo like sampling in order to estimate the performance of an exemplary control task.

We will see that uncorrected uncertainties diminish the control performance in varying degrees but in general are acceptable with respect to usual accuracy demands in the mission requirements.

5.4.1 Sources of Errors

In this subsection a small overview of the different sources of uncertainties in the system shall be given. Roughly, these can be grouped into three sources of errors, mainly ones due to errors in the actuation, the model of the system dynamics and measurement uncertainties. In contrast to performance weakness, for example, in the controller itself, these kind of errors can be seen as nondeterministic and random. Because of the non-linearity and complexity of the system, it becomes difficult to predict to which degree the combination of uncertainties affects the control performances.

Thruster Specific Errors

First, we can identify errors specific to the thruster actuation. Some of these were already mentioned in Section 2.1. A, not-necessarily complete, list can be given in the following

1. Thruster misalignment: The pose of a thruster is specified by its three position values and its orientation given by azimuth and elevation angles. The position can deviate by a few millimeters due to construction, while the orientation is uncertain with respect to around one degree for both azimuth and elevation angles.

2. Relative frame error of geometrical frame: We recall that the thrusters are defined in the geometrical frame, while the control wrenches are not. The geometrical frame can also be erroneous in translation and rotation with respect to the frame in which the control wrenches are generated.

3. Deviation from nominal thrust and transition time of thruster: As already introduced, the applied thrust is not as constant as assumed. It jitters around its nominal value and can also have a constant bias. Together with the required transition time between on and off states, this means that not the nominal wrench of the thruster is applied for all times.

4. Chance of misfiring: For each transition of the thrusters, there is also a chance that a desired thruster does not fire at all due to a problem with the fuel distribution system. In the worst case, an undetected defective valve or thruster can lead to a significant uncertainty.

Dynamic Errors

The structural parts of the space robot are also not ideal. Both the spacecraft body and the robot arm are prone to fabrication tolerances. The most important effects on the dynamical behavior are listed here shortly.

1. Error in Mass: Due to manufacturing tolerances, the masses of each part deviate slightly from their nominal values. More importantly though, the mass of the body significantly changes depending on the consumed fuel.

The system behavior might deteriorate if the control gains are not adapted accordingly. While one can estimate the remaining fuel and the corresponding mass, this estimation also underlies an error.

2. Error in location of COM: Also the COM location of each of the parts has some tolerances. In addition, the potential movement of the liquid fuel can heavily change the COM of the base and thus the dynamical behavior of the complete system.

3. Error in Inertia: Not only the location of the COM is uncertain, but also the mass distribution and therefore the inertia values themselves.

Measurement Uncertainties

In space, it becomes difficult to accurately measure the velocity and pose of the spacecraft. Usually, the low measurement rates require observer models, which fuse different sensor input and make the determination of the states more robust.

Consequently, the achievable accuracy of the given control task is inherently limited by the quality of the state estimations.

5.4.2 Monte Carlo Sampling

In order to be able to estimate the influence of the single uncertainties in the model, at first each uncertainty was modeled on its own and finally also different combinations of them together. Multiple simulations were carried out, where the performance of a given control task was measured with different kinds of setups of active uncertainties.

As indicated before, it is assumed that, for example, the deviation of the real thruster alignment from its nominal values is unknown. Heuristics are used to determine in which bounds the maximal deviation can occur. Assumptions on the probability density of each uncertainty have to be made. For most of the uncertainties, the obvious choice is to draw from a Gaussian distribution with the nominal value of the model being equal to its mean.

It is not clear how the control performance is affected by different combinations of uncertainties. However, for a configuration of 24 thrusters, the potential misalignment of each thruster alone leads to 120 erroneous parameters, which results in an infinite number of possible combinations.

We are mainly interested in the boundaries of the control performance and its likelihood. Therefore, we fall back to a Monte Carlo Sampling like approach, where a limited amount of samples is used to approximate this incalculable properties of the performance distribution.

The general approach for all scenarios of active uncertainties is shortly summarized in the following.

1. Specify which sets of uncertainties are active
2. Specify the probability distribution of the uncertain parameters
3. Sample a number of Monte Carlo samples for all model properties
4. Carry out a number of simulations specified by the number of samples for a given control task
5. Evaluate the deviation of the control performance with respect to different metrics

5.4.3 Selected Results from Simulation in Robot Simulator

The control task and overall setup still remains the same compared to Section 5.2. Only now, the control task is conducted with some uncertainties present. Multiple different scenarios with different combinations and sources of errors were created. In this section a short overview of the main results will be shown. Note that this consideration is still only a shallow introduction and cannot cover all sources of uncertainties.

The structure of the scenarios is as follows. A summary of the described uncertainty values can be found in Table 5.1.

1. Scenario 1 (only thruster misalignment): For each thruster, we have Gaussian distributions with a STD of $\sigma = 0.5^\circ$ for the thruster orientation and also the position errors are Gaussian with $\sigma = 0.5mm$. Additionally, a potential translation between COM frame of the base and geometrical thruster frame according to a Gaussian with $\sigma = 2.5cm$ is present. This means that not only the single thrusters but also the complete configuration is misaligned with respect to the base frame.

2. Scenario 2 (uncertainties in base): The mass value is drawn from a uniform distribution with $\pm 5\%$ deviation from the nominal value of 300kg. The inertia values are changed according to the changed mass, but the result has an additional Gaussian uncertainty of $\sigma = 2.5\%$. Lastly, also the position of the COM of the base is deviating from its nominal value according to $\sigma = 2.5cm$.

3. Scenario 3 (uncertainties in robot arm): A similar approach to Scenario 2 is conducted here, only for the seven links of the robot arm instead of the base. We use the same values for the COM position, mass and inertia uncertainties for each link. Though, it has to be noted that the percentage mass uncertainty in the links is this time drawn from a Gaussian and then added to the nominal value. A big mass change in the base is more probable due to the consumption of (some) fuel, while the links have a specified nominal mass which only changes due to manufacturing error.

4. Scenario 4 (uncertainties in gripper): We apply the same uncertainty setup from Scenario 3, only for the gripper. This means that only the mass, inertia and COM of the gripper is erroneous.

5. Scenario 5 (combination): We use a combination of all the previous scenarios together. Therefore, all the previously described uncertainties will be active at once.

6. Scenario 6 (sensor noise): In this step we are not modeling the errors occurring for different sensors, but we directly put uncertainties in a measurement model for position and orientation of the base and the end-effector. These can be expressed as errors in the frames with respect to the target frame and have the quality of realistic observer output. For our qualitative analysis, the generic magnitude of sensing uncertainties in our space robot environment are sufficient. Obviously, a more detailed analysis requires further definitions and a detailed differentiation of sensors. For example, raw Lidar outputs would give more noisy data. The measured states are prone to Gaussian noise. Without getting to detailed here, the standard deviations of all measurement errors for position and orientation are in the area of 0.3°, $0.3^\circ/s$, $3mm$ and $2\frac{mm}{s}$ respectively for both base and end-effector.

7. Scenario 7-11: These scenarios are equal to scenarios 1 to 5 only with the addition of sensor noise which was described in Scenario 6

For each sample, the deviation values are drawn from their respective distribution and the control task is carried out. Overall a sum of 200 samples was conducted for each scenario. We can see that the number of samples might be insufficient to fully obtain the worst-case performance boundaries. For example, for Scenario 1 we draw in total a number of 123 uncertain parameters, which are independent from each other. For definitive statements regarding the quality bounds of the resulting control a more analytical approach is required.

Our setup of the scenarios allows us to oppose the effect of measurement noise to the other sources of uncertainty. In Section 5.3, we introduced the quality measures, which are also used in the following in order to visualize the approximated error bounds.

In order to reduce the number of required plots and keep the readability high, only the most important subset of scenarios will be plotted in the graphs. We are mainly interested in the static performance after convergence. Therefore, we will only start to investigate after convergence of the end-effector, which can be observed after approximately six seconds. The time period of the first six seconds where mainly the end-effector task is carried out can be disregarded.

For all the following plots, we will keep the same color correspondences. The reference baseline is given by the blue color and has no uncertainties present. Scenario 1 is visualized in red, while Scenario 5 is plotted in green. Scenario 6 is visualized in orange and Scenario 11 is given in purple.

Parameter	Distribution	Deviation (min-max or STD)	% nominal
Thruster displacement (x,y,z)	Gaussian	0.5 mm	n.a.
Thruster misalignment (χ,γ)	Gaussian	0.5 deg	n.a.
Displacement geometrical to base frame (x,y,z)	Gaussian	2.5 cm	n.a.
Spacecraft mass	Uniform	$\pm$15 kg	5%
Spacecraft inertia uncertainty (additional to mass scaling)	Gaussian	2.5 %	2.5%
Spacecraft CoM (x,y,z)	Gaussian	2.5 cm	n.a.
Link mass uncertainty (all 7)	Gaussian	2.5 %	2.5%
Link inertia uncertainty (all 7) (additional to mass scaling)	Gaussian	2.5 %	2.5%
Link CoM (x,y,z) (all 7)	Gaussian	2.5 cm	n.a.
Gripper mass uncertainty	Gaussian	2.5 %	2.5%
Gripper inertia uncertainty (additional to mass scaling)	Gaussian	2.5 %	2.5%
Gripper CoM (x,y,z)	Gaussian	2.5 cm	n.a.
Model measurement errors	Gaussian	3mm	n.a.
(for position and orientation	Gaussian	2mm/s	n.a.
of end-effector with respect	Gaussian	0.3 deg	n.a.
to target frame)	Gaussian	0.3 deg/s	n.a.

Table 5.1: Summarized list of modeled uncertainties (list of model measurement errors is not exhaustive)

End-Effector Error

Once again, the most important reference value is given by the quality of performance for the end-effector control task.

At first, we will look at some time responses for selected scenarios and compare them to the baseline value. For this we select only one random sample in each plotted scenario. Figure 5.6 delivers two first insights. The non-zero momentum leads to a constant correction procedure. In contrast to the simple one-dimensional case, no constant periodic limit cycle can be observed but only a more complex output. Nevertheless, we use the term limit cycle to stress the relation to its origin. The time interval of the limit cycles is equal to how often a correction of position or orientation can be observed.

As a second result, we can observe that the introduction of sensor noise has a significant influence, while the effect due to the other uncertainties remains relatively low. Thus, it comes to no surprise that in Scenario 11 the error seems to be pre-dominated by the measurement noise. Though, also with sensor noise the system can stay within the range of the measurement errors.

We recall that the position and orientation of the end-effector compared to the target is measured with a Gaussian error with standard deviation of $3mm$ and $0.3°$.

If we only take the minimal and maximal error after convergence for each sample, we can create a histogram from all samples and verify these first results. For example the minimal translational error in x-direction $e_{min,x}$ is obtained by looking at all x-position errors e_x in the simulations after 10s simulation time.

$$e_{min,x} = min(e_x[10s, ..., 20s]) \tag{5.7}$$

For example for Figure 5.6 and Scenario 11 we have $e_{min,x} \approx -2.8mm$. In a similar fashion this can be also carried out for all the other parameters of interest.

Note that the convergence time was selected to be 10s in this simulation set and not 6s which would also be valid for most of the samples. This guarantees us that a longer convergence time for some samples doesn't affect our results. We are capable of identifying the variance of the control quality. Figures 5.7 and 5.8 provide insights about the error distribution. We can see that the quality of the end-effector task is dependent on the sampled uncertainties. Not all kinds of uncertainties lead to a critical degeneration of the control.

Mainly the measurement noise leads to a significant increase of the error bounds. Thruster misalignments or uncertainties in the inertia have a significantly lower effect. The effect is so low that the result from Scenario 11 is dominated by the measurement noise. It is hardly possible to identify a difference compared to Scenario 6, where solely the measurement noise is present. The mean is centered around position errors of around 2-3mm, while the roll-pitch-yaw angles deviate by around $0.3° - 0.6°$. It has to be noted though that the control quality is still sufficiently high in order to achieve the mission requirements.

For all entries in the position and orientation vectors, the error boundaries seem to be very similar. Though, when looking at Scenarios 1 and 5, we can see that the initial values and the general control task still seem to influence the output. For these scenarios, we can see that, for example, the position difference along the z-axis is generally higher than for the other axes. Note that the baseline plot is often very close to Scenario 1 and therefore covered in the plots.

An important thing to keep in mind for all the following histograms is that, in order to allow the plotting of all scenarios in one single figure, the interval width for each histogram is set to be variable but the number of bins is fixed to 25. This means that the error intervals, used to determine the number of samples within a bin, are adapted accordingly. Therefore, these are also different for each scenario depending on the resulting overall spread.

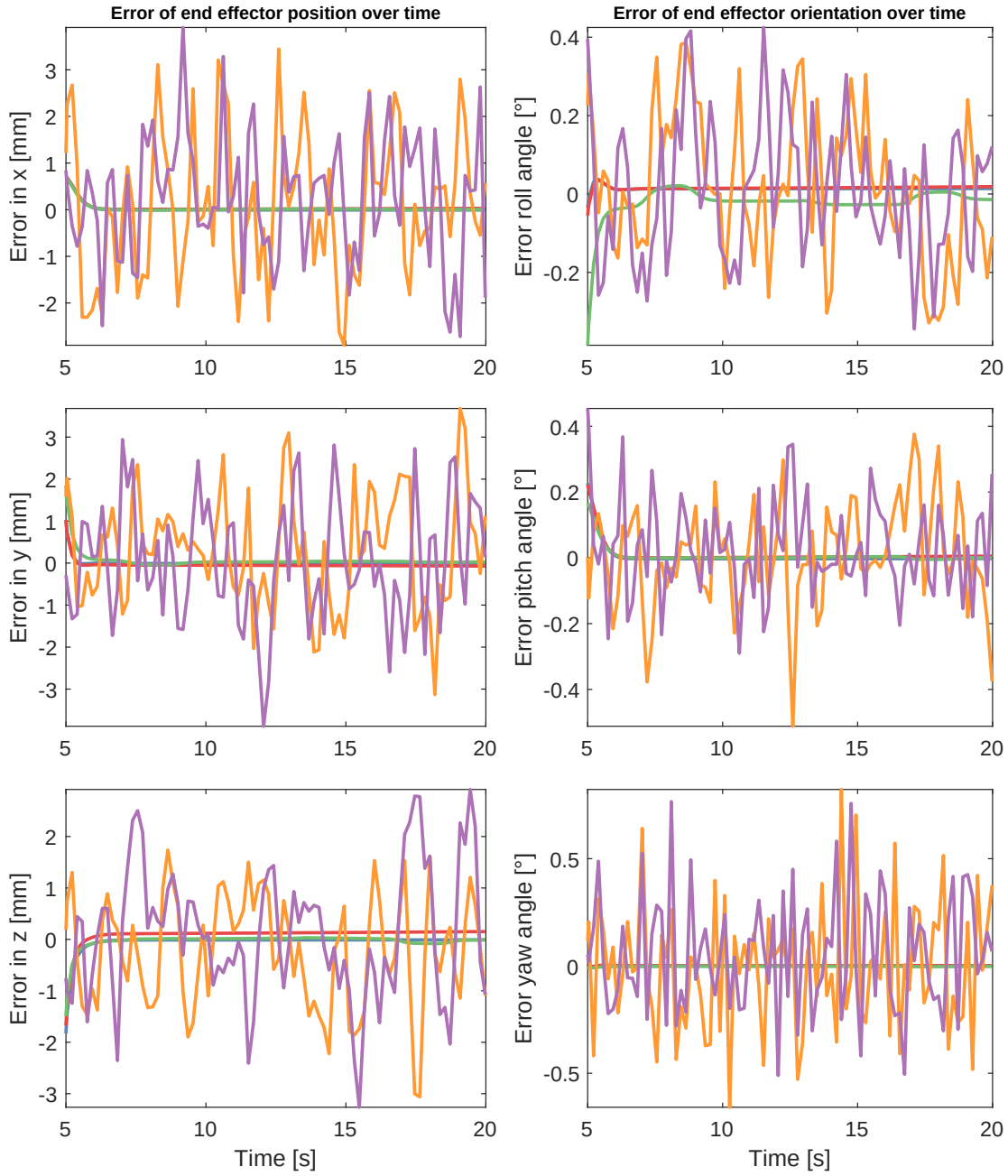


Figure 5.6: Uncertainties: Position and orientation error in end-effector over time for one sample (for Baseline(blue), Scenario 1(red), Scenario 5 (green), Scenario 6 (orange) and Scenario 11 (purple))

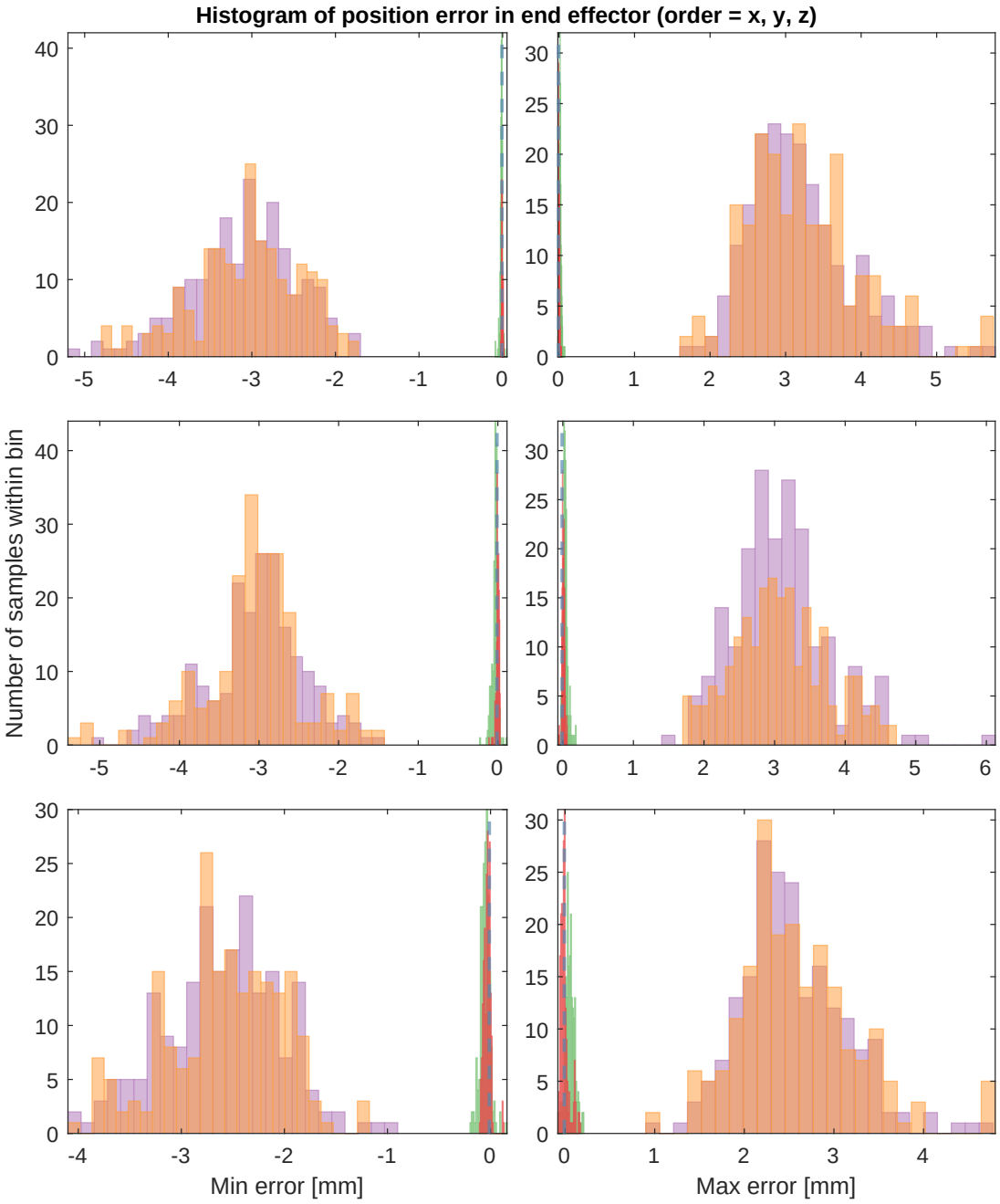


Figure 5.7: Uncertainties: Histogram of position error in end-effector for 200 samples (for Baseline(blue), Scenario 1(red), Scenario 5 (green), Scenario 6 (orange) and Scenario 11 (purple))

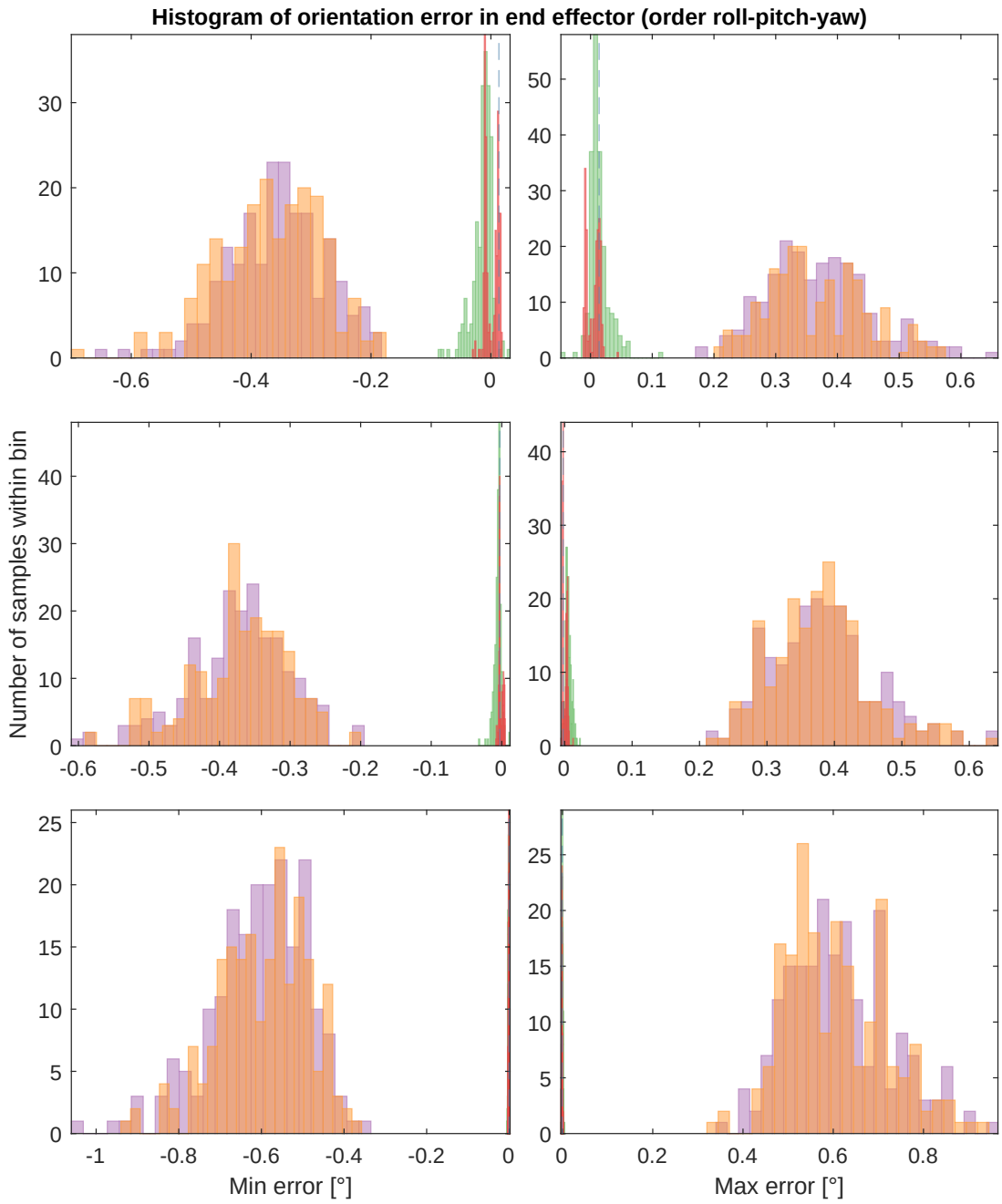


Figure 5.8: Uncertainties: Histogram of orientation error in end-effector for 200 samples (for Baseline(blue), Scenario 1(red), Scenario 5 (green), Scenario 6 (orange) and Scenario 11 (purple))

Error in COM position of Complete System

In Scenario 2, an uncertainty of the COM position of the base is introduced. Consequently, this uncertainty also influences the COM of the complete system. This is why we have an initial error at time zero for scenarios where the real COM of the sup-parts is changed. In the plots, this is the case for Scenarios 5 and 11.

However, the starting error in the COM can only be successfully removed over time when no additional sensor noise is present (Scenario 5). The COM error is asymptotically reduced to nearly zero for all three directions. In contrast, Scenario 11 is barely able to keep the initial error in the COM of the complete system stable, but cannot reduce it. This becomes apparent in Figure 5.9, where the time responses for one random sample were plotted. We see that a significant COM offset remains in Scenario 11, which is nearly identical to the initial error. Also Scenario 6 has an observable jittering around the desired position.

We can also see that the scenarios with sensor noise are significantly more active. The slope of the COM position is corrected relatively often, which can be also observed as a high rate limit cycle. As we will analyze later, this results in a high fuel consumption.

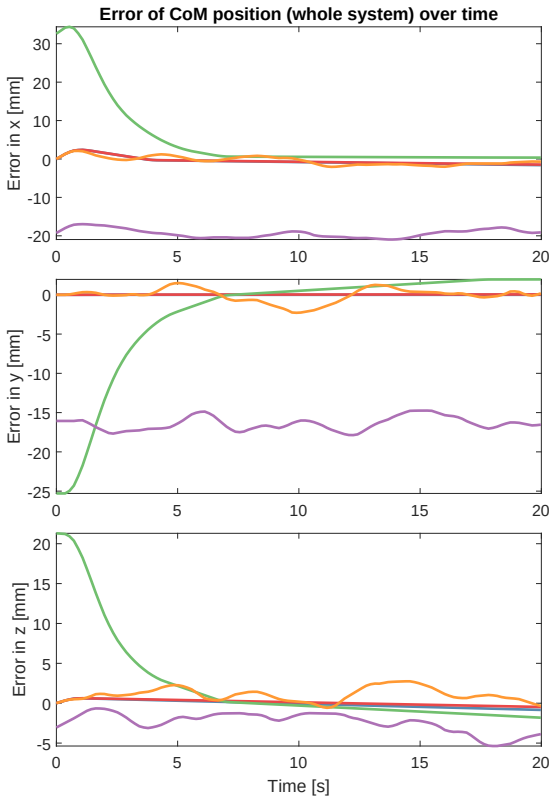


Figure 5.9: Uncertainties: Error in position of COM of complete system over time (for Baseline(blue), Scenario 1(red), Scenario 5 (green), Scenario 6 (orange) and Scenario 11 (purple))

The previous results from the observation of one sample can be also verified for the complete number of samples, depicted in the histogram in Figure 5.10.

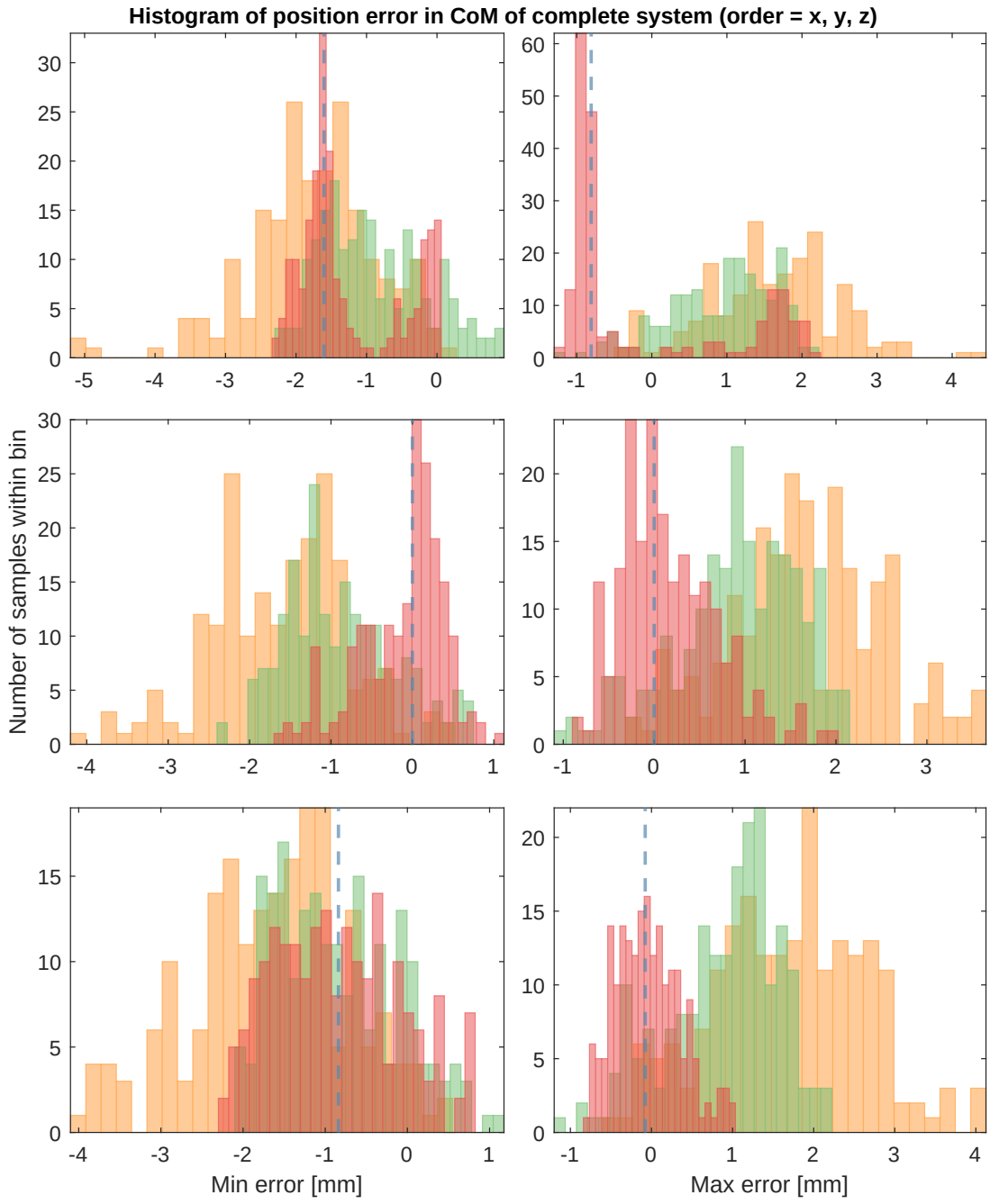


Figure 5.10: Uncertainties: Histogram of position error of COM of complete system for 200 samples (for Baseline(blue), Scenario 1(red), Scenario 5 (green) and Scenario 6 (orange)

Having only misalignments in the thrusters seems to have little impact on the result (Scenario 1), while an initial error of the COM position of the complete system also leads to a higher final error after convergence (Scenario 5). Note that also the baseline result, obtained without having uncertainties, always has non-zero errors due to the non-zero momentum in an environment without friction. The error bound spanned by min and max errors is very small though, which means that the error often appears to be zero.

As soon as sensor noise is introduced (Scenario 6), the COM error increases more significantly. The result from Scenario 11 is not plotted. As discussed before, the initial offset due to the COM uncertainty of base or links cannot be resolved in this scenario. Thus, the sample errors are approximately equal to the sampled initial error. This means that the histogram of Scenario 11 is spread out and would reduce the visibility of the entire plot, while not introducing any new insights.

Angular Momentum around Centroid

By plotting the angular momentum of the system in Figure 5.11 , we can verify that the introduction of sensor noise leads to a persistent actuation of the thrusters (Scenario 6 and 11). In these scenarios the amplitude of the limit cycles is also significantly larger than for the others. In addition, we can observe that the quality of the control is spread out more for Scenarios 6 and 11 across all samples. The introduction of thruster misalignments in Scenario 1 introduces only a small additional deviation. Though, with uncertainties in the dynamics models, as present in Scenario 5, we observe increased maximal momentum values.

Fuel

Due to uncertainties, not only the accuracy of the control task may suffer, but it will then also be fulfilled less efficiently. The previously discussed effects can be also seen in the fuel consumption depicted by Figure 5.12. As mentioned before, the addition of a COM displacement in the base (Scenario 5) automatically enforces corrections of the overall COM. Depending on the sampled displacement value, this will require more or less actuation. This is why in the left plot of figure 5.12, we observe a large increase in fuel consumption at the start of the simulation. When using thrusters, the momentum never reaches zero and thus the limit cycle continuously increases the fuel consumption. However, if the thrusters are misaligned and due to the previously discussed dead-zone, a high momentum cannot be reduced efficiently and results in faster limit cycles.

Once again we can observe that the sensor noise significantly diminishes the control performance. Normally, after some time, the fuel consumption stabilizes for this setup. Then, only the limit cycle actuation leads to a slight increase due to short firing corrections. However, the sensor noise leads to a permanent actuation for all times so that fuel is consumed nearly linearly.

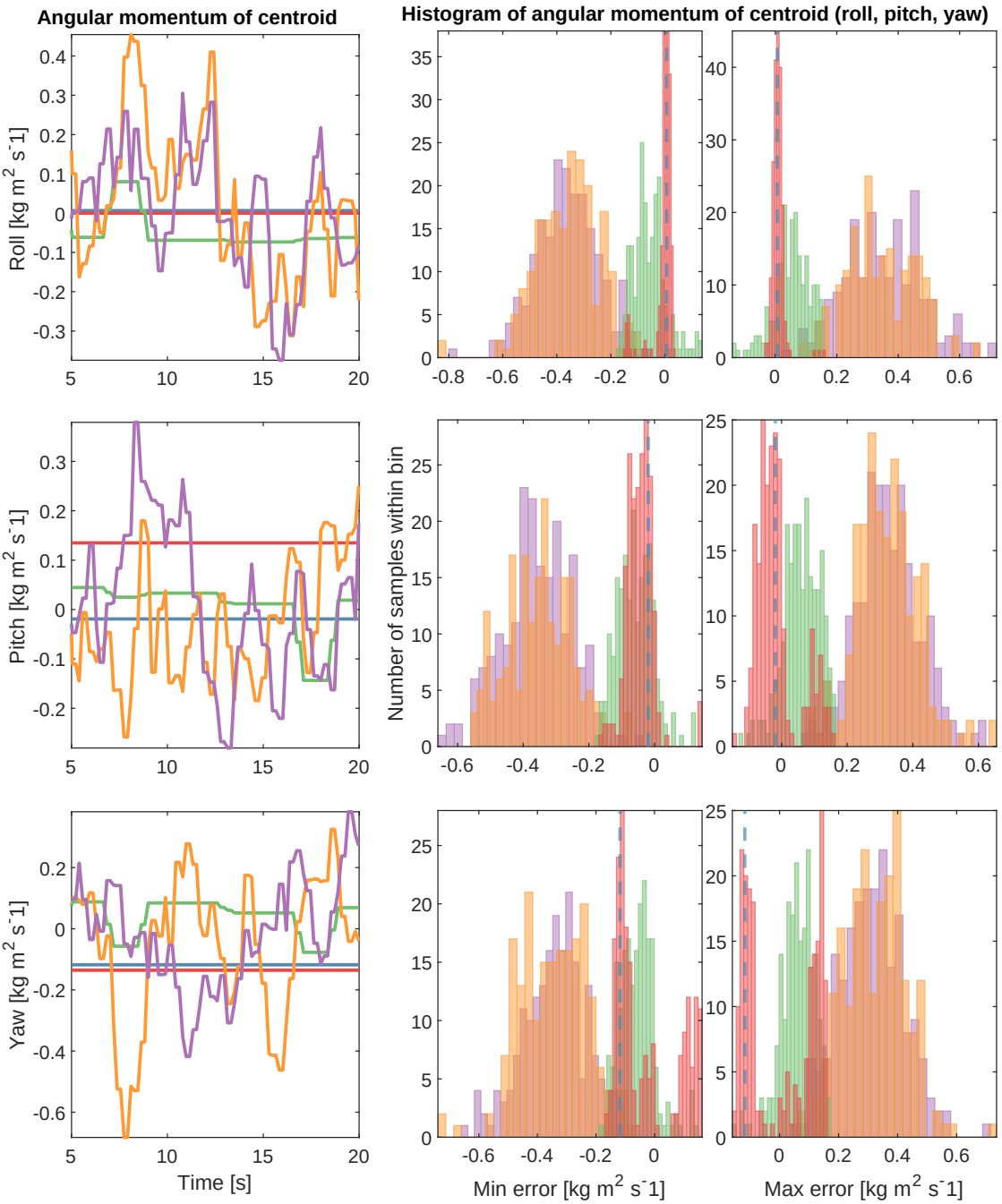


Figure 5.11: Uncertainties: Angular momentum over time and histogram for 200 samples (for Baseline(blue), Scenario 1(red), Scenario 5 (green), Scenario 6 (orange) and Scenario 11 (purple))

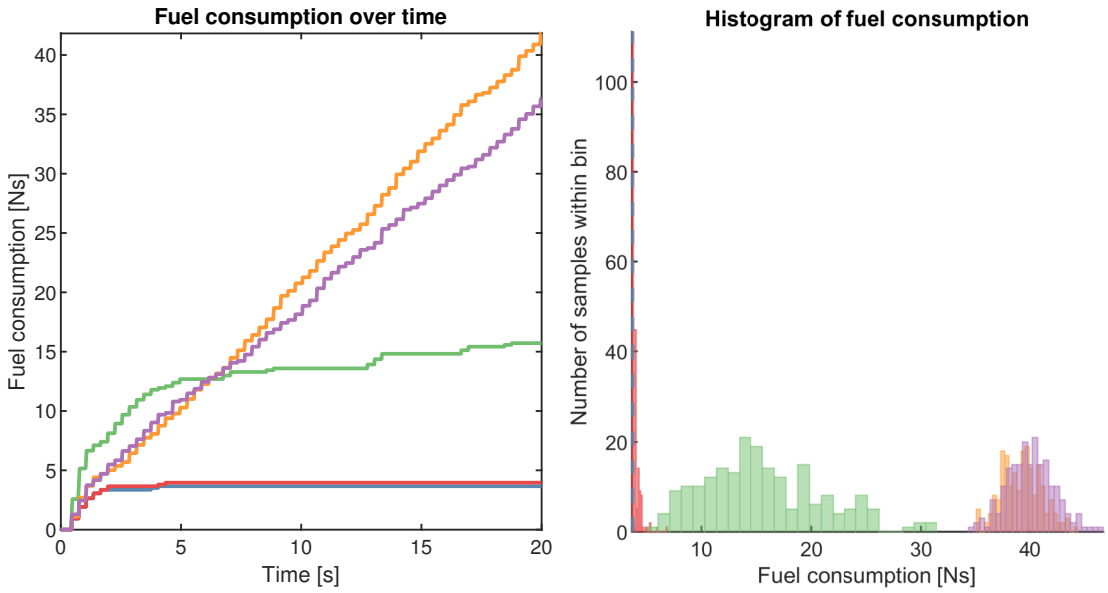


Figure 5.12: Uncertainties: Angular momentum over time and histogram for 200 samples (for Baseline(blue), Scenario 1(red), Scenario 5 (green), Scenario 6 (orange) and Scenario 11 (purple))

5.5 Conclusion

Without any model or thruster uncertainties and for the lack of sensor noise, we can verify the high system accuracy when using Mixed Integer Linear Programming assignments in the exemplary control task.

As a main result of this chapter, we can conclude that the presence of different sources of uncertainties affects the performance and also the thruster actuation only marginally in general. However, different kinds of uncertainties can have a varying effect. For our selection, we observed that the sensor noise has the largest influence. All other uncertainties become irrelevant compared to this parameter. All performance parameters were within the accuracy bounds required for typical mission requirements. The main problem lies within fast limit cycles and the resulting high fuel consumption which can be observed mainly with sensor noise.

It might be beneficial to develop a robust observer model, which resolves the main problem of rapidly changing measurements. In our model, each sensor input is contaminated by a Gaussian deviation, which potentially results in a constant correction of the previous control time step. For the thrusters, this means that we quickly consume a lot of fuel. The problem might be practically mitigated by employing a deadzone to inhibit thruster firing withing thresholds defined by the noise level on the momentum.

Nevertheless, also the influence of thruster misalignments can be important with respect to the layout of the thruster configuration and the selected assignment method.

A very accurate assignment method, which requires additional computational resources, might become obsolete under the presence of misalignments. Though, it is still an open question of how these different error sources interact with each other. In the conducted simulation, on average both errors did not seem to be additive.

6

Conclusion and Outlook

It was shown that a divide-and-conquer approach can help to isolate different sub-tasks and identify the different leverage points for the optimization of the thruster-based system. The effects of different choices within the pipeline were laid out. With this knowledge limitations can be identified and adapted early on. For each sub-task, state of the art references were given and additionally new approaches were developed.

Overall, four main topics could be identified and were the focus of this work.

First, the thruster actuation was investigated. It has to be implemented in a way so that the thrusters mimic a continuous, ideal wrench actuator which is only bounded due to the maximal thrust. The assignment and modulation can be tuned so that we can get a performance that reaches the maximal envelope, set up by the physical limits of the thruster configuration.

This work proposed two optimization methods, one based on Mixed Integer Programming, the other one on Quadratic Programming, which both showed a lot of potential. The Mixed Integer Programming method allows to include more thruster properties, which results in higher accuracy and overall better assignments. A big advantage here is also the high flexibility, which allows a customization depending on specific requirements. The Quadratic Programming formulation guarantees good assignments even for control wrenches outside the controllability envelope.

However, limitations on the computational resources might prohibit their usage. Further works may address improvements in order to limit the complexity of these assignment methods. It might be possible to combine the proposed methods with precomputed table methods. This is already applied for Linear Programming formulations. Here, the developed notion of counter-thrust might also prove to be useful in the future.

Not only the assignment but also the most frequently used modulation were analyzed and it was shown how one can tune the modulators and also how to relate the Pulse-Width Pulse-Frequency modulator with a Pulse-Width Modulation.

Second, the thruster configuration itself has to be analyzed. It was shown that the theory of positive linear spaces and analogies of grasp planning and swimming robots are useful in order to obtain new insights.

In the future more in-depth exploitation of these areas of research might prove to be meaningful to further advance thruster configurations for space robots.

A possible approach for the development of different configurations, using a different number of thrusters, was discussed. The presented framework uses reasonable assumptions in order to simplify the problem. Though, further research is still required for how to optimally place thrusters on a given arbitrary spacecraft geometry and how this placement changes with the number of used thrusters. The problem here mainly lies within the high dependency on specific project requirements and that it is hard to compare different configurations.

This leads to an increased requirement for computationally efficient and meaningful performance indexes to compare candidate configurations. Proposals for these, which can be extended in the future, were given in this work.

Third, the selected control type has to ideally match the nature of the actuation and the system. In a sense, the control has to be tuned in a way that ensures that only efficient control commands are generated. This task is heavily reliant on the project requirements.

This work heavily focused on simplified one-dimensional settings, where direct tuning recommendations could be given. Though, also one exemplary realistic real world maneuver was simulated. The previously developed theoretic results could be observed. In a next step, it is obvious that the conducted simulations still have to be verified on the hardware-in-the-loop testing facility.

Lastly, high accuracy control cannot be obtained if the uncertainties cannot be reduced to a minimum. The high effort in the best thruster actuation methods becomes meaningless if the uncertainties in measurement and configuration are too large.

For a reasonable selection of uncertainties, this work showed that mainly the sensor noise has an impact. However, the control performance even under these kind of uncertainties was still acceptable. Though, performant observer models might have to be applied mainly in order to reduce the observed increase in fuel consumption. In the literature (e.g. [76] and [77]), extended Kalman Filters and observers are already available and can be applied with respect to this task.

Additionally, a reliable way of obtaining the control performance interval for all possible combination of uncertainties has to be found. As pointed out in this work, there exist combinations of uncertainties with different levels of significance. Generally, a more structured approach might be required, which not only approximates but calculates the boundaries of the control performance. Reachability analysis and set-based methods come to mind.

List of Figures

List of Tables

Bibliography

[1] P. Acquatella, "Development of automation & robotics in space exploration," 2007.

[2] G. Rekleitis and E. Papadopoulos, "On-orbit cooperating space robotic servicers handling a passive object," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 51, no. 2, pp. 802–814, 2015, ISSN: 0018-9251. DOI: 10.1109/TAES.2014.130584.

[3] A. Flores-Abad, O. Ma, K. Pham, and S. Ulrich, "A review of space robotics technologies for on-orbit servicing," *Progress in Aerospace Sciences*, vol. 68, pp. 1–26, 2014, ISSN: 03760421. DOI: 10.1016/j.paerosci.2014.03.002.

[4] A. Ellery, J. Kreisel, and B. Sommer, "The case for robotic on-orbit servicing of spacecraft: Spacecraft reliability is a myth," *Acta Astronautica*, vol. 63, no. 5-6, pp. 632–648, 2008, ISSN: 00945765. DOI: 10.1016/j.actaastro.2008.01.042.

[5] K. Landzettel, C. Preusche, A. Albu-Schaffer, D. Reintsema, B. Rebele, and G. Hirzinger, "Robotic on-orbit servicing - dlr's experience and perspective," in *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, IEEE, 9.10.2006 - 15.10.2006, pp. 4587–4594, ISBN: 1-4244-0258-1. DOI: 10.1109/IROS.2006.282164.

[6] W. Fehse, *Automated rendezvous and docking of spacecraft*, ser. Cambridge aerospace series. Cambridge: Cambridge University Press, 2003, vol. 16, ISBN: 978-0-521-82492-7. DOI: 10.1017/CBO9780511543388.

[7] S. A. A. Moosavian and E. Papadopoulos, "Free-flying robots in space: An overview of dynamics modeling, planning and control," *Robotica*, vol. 25, no. 5, pp. 537–547, 2007, ISSN: 0263-5747. DOI: 10.1017/S0263574707003438.

[8] S. Dubowsky and E. Papadopoulos, "The kinematics, dynamics, and control of free-flying and free-floating space robotic systems," *IEEE Transactions on Robotics and Automation*, vol. 9, no. 5, pp. 531–543, 1993, ISSN: 1042296X. DOI: 10.1109/70.258046.

[9] E. Papadopoulos and S. Dubowsky, "Coordinated manipulator/spacecraft motion control for space robotic systems," in *Proceedings. 1991 IEEE International Conference on Robotics and Automation*, IEEE Comput. Soc. Press, 9-11 April 1991, pp. 1696–1701, ISBN: 0-8186-2163-X. DOI: 10.1109/ROBOT.1991.131864.

[10] A. M. Giordano, G. Garofalo, M. de Stefano, C. Ott, and A. Albu-Schaffer, "Dynamics and control of a free-floating space robot in presence of nonzero linear and angular momenta," in *2016 IEEE 55th Conference on Decision and Control (CDC)*, IEEE, 12.12.2016 - 14.12.2016, pp. 7527–7534, ISBN: 978-1-5090-1837-6. DOI: 10.1109/CDC.2016.7799432.

[11] A. M. Giordano, A. Dietrich, C. Ott, and A. Albu-Schäffer, "Coordination of thrusters, reaction wheels, and arm in orbital robots," *Robotics and Autonomous Systems*, vol. 131, p. 103 564, 2020, ISSN: 09218890. DOI: 10.1016/j.robot.2020.103564.

[12] E. Oland, "Modeling and attitude control of satellites in elliptical orbits," in *Applied Modern Control*, L. Anh Tuan, Ed., IntechOpen, 2019, ISBN: 978-1-78984-826-7. DOI: 10.5772/intechopen.80422.

[13] F. Curti, M. Romano, and R. Bevilacqua, "Lyapunov-based thrusters' selection for spacecraft control: Analysis and experimentation," *Journal of Guidance, Control, and Dynamics*, vol. 33, no. 4, pp. 1143–1160, 2010, ISSN: 0731-5090. DOI: 10.2514/1.47296.

[14] J. D. Biggs and H. Fournier, "Neural-network-based optimal attitude control using four impulsive thrusters," *Journal of Guidance, Control, and Dynamics*, vol. 43, no. 2, pp. 299–309, 2020, ISSN: 0731-5090. DOI: 10.2514/1.G004226.

[15] A. E. Bryson, *Control of Spacecraft and Aircraft*. Princeton: Princeton University Press, 1994, ISBN: 978-1-4008-8003-4.

[16] W. G. d. Santos, E. M. Rocco, T. Boge, H. Benninghoff, and F. Rems, "Multi-objective optimization applied to real-time command problem of spacecraft thrusters," *Journal of Spacecraft and Rockets*, vol. 52, no. 5, pp. 1407–1416, 2015, ISSN: 0022-4650. DOI: 10.2514/1.A33178.

[17] M. J. Sidi, *Spacecraft dynamics and control: A practical engineering approach*, [Elektronische Ressource], ser. Cambridge aerospace series. Cambridge: Cambridge Univ. Press, 2014, vol. 7, ISBN: 0-521-55072-6. DOI: 10.1017/CBO9780511815652.

[18] P. A. Servidia and R. S. Pena, "Spacecraft thruster control allocation problems," *IEEE Transactions on Automatic Control*, vol. 50, no. 2, pp. 245–249, 2005, ISSN: 0018-9286. DOI: 10.1109/TAC.2004.841923.

[19] M. Bodson, "Evaluation of optimization methods for control allocation," *Journal of Guidance, Control, and Dynamics*, vol. 25, no. 4, pp. 703–711, 2002, ISSN: 0731-5090. DOI: 10.2514/2.4937.

[20] F. Ankersen, S.-F. Wu, A. Aleshin, A. Vankov, and V. Volochinov, "Optimization of spacecraft thruster management function," *Journal of Guidance, Control, and Dynamics*, vol. 28, no. 6, pp. 1283–1290, 2005, ISSN: 0731-5090. DOI: 10.2514/1.11531.

[21] A. Smith and D. Seo, "Spacecraft thruster distribution matrix for precision 6dof control," in *AIAA SPACE and Astronautics Forum and Exposition*, Reston, Virginia: American Institute of Aeronautics and Astronautics, 9122017, p. 461, ISBN: 978-1-62410-483-1. DOI: 10.2514/6.2017-5203.

[22] H. Johansson, "Optimal 'thruster actuation in high precision attitude and orbit control systems," Master's thesis, Lulea University of Technology, Lulea, Sweden, 2005.

[23] J. Virnig and D. Bodden, "Multivariable control allocation and control law conditioning when control effectors limit," in *Guidance, Navigation, and Control Conference*, Reston, Virigina: American Institute of Aeronautics and Astronautics, 81994. DOI: 10.2514/6.1994-3609.

[24] R. Eberhardt and D. Ward, "Indirect adaptive flight control of a tailless fighter aircraft," in *Guidance, Navigation, and Control Conference and Exhibit*, Reston, Virigina: American Institute of Aeronautics and Astronautics, 8091999. DOI: 10.2514/6.1999-4042.

[25] C. Gazzino, C. Louembet, D. Arzelier, N. Jozefowiez, D. Losa, C. Pittet, and L. Cerri, "Integer programming for optimal control of geostationary station keeping of low-thrust satellites," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 8169–8174, 2017, ISSN: 24058963. DOI: 10.1016/j.ifacol.2017.08.1264.

[26] J. Hooker, *Logic-based methods for optimization: Combining optimization and constraint satisfaction*, ser. Wiley-Interscience series in discrete mathematics and optimization. New York: John Wiley & Sons, 2000, ISBN: 9780471385219. DOI: 10.1002/9781118033036.

[27] H. P. Williams, *Logic and Integer Programming*, ser. International Series in Operations Research & Management Science. Boston, MA: Springer-Verlag US, 2009, vol. 130, ISBN: 978-0-387-92280-5. DOI: 10.1007/978-0-387-92280-5.

[28] H. P. Williams, *Model building in mathematical programming*, Fifth Edition. Chichester: John Wiley & Sons Ltd, 2013, ISBN: 978-1-118-44333-0.

[29] A. Furmanek, *Ilp part 4 – comparisons*, 2015.

[30] J. D. Camm, A. S. Raturi, and S. Tsubakitani, "Cutting big m down to size," *Interfaces*, vol. 20, no. 5, pp. 61–66, 1990, ISSN: 0092-2102. DOI: 10.1287/inte.20.5.61.

[31] F. Martel, "Optimal 6 axis command of a space vehicle with a precomputed thruster selection catalogue table," in *18th International Symposium on Space Flight Dynamics*, ser. ESA Special Publication, vol. 548, 2004, p. 595.

[32] N. Silva, F. Martel, and P. Delpy, "Automated transfer vehicle thrusters selection and management function," in *Guidance, Navigation and Control Systems*, ser. ESA Special Publication, vol. 606, 2006, p. 28.

[33] B. Wie, *Space vehicle dynamics and control*, 2. ed., ser. AIAA education series. Reston, Va: American Inst. of Aeronautics and Astronautics, 2008, ISBN: 978-1-56347-953-3.

[34] F. Ankersen, *Guidance, navigation, control and relative dynamics for spacecraft proximity maneuvers: Ph. D. thesis*, 1. edition. Aalborg: Aalborg University, 2011, ISBN: 978-87-92328-72-4.

[35] G. Song, N. V. Buck, and B. N. Agrawal, "Spacecraft vibration reduction using pulse-width pulse-frequency modulated input shaper," *Journal of Guidance, Control, and Dynamics*, vol. 22, no. 3, pp. 433–440, 1999, ISSN: 0731-5090. DOI: 10.2514/2.4415.

[36] B. N. Agrawal, R. S. Mcclelland, and G. Song, "Attitude control of flexible spacecraft using pulse-width pulse-frequency modulated thrusters," *Space Technology*, vol. 17, no. 1, pp. 15–34, 1997.

[37] Q.-l. Hu and G.-f. Ma, "Flexible spacecraft vibration suppression using pwpf modulated input component command and sliding mode control," *Asian Journal of Control*, vol. 9, no. 1, pp. 20–29, 2007, ISSN: 15618625. DOI: 10.1111/j.1934-6093.2007.tb00300.x.

[38] B. N. Agrawal, R. S. Mcclelland, and G. Song, "Attitude control of flexible spacecraft using pulse-width pulse-frequency modulated thrusters," *Space Technology-Kedlington*, vol. 17, no. 1, pp. 15–34, 1997.

[39] G. Arantes, L. S. Martins-Filho, and A. C. Santana, "Optimal on-off attitude control for the brazilian multimission platform satellite," *Mathematical Problems in Engineering*, vol. 2009, no. 12, pp. 1–17, 2009, ISSN: 1024-123X. DOI: 10.1155/2009/750945.

[40] A. Khosravi and P. Sarhadi, "Tuning of pulse-width pulse-frequency modulator using pso: An engineering approach to spacecraft attitude controller design," *Automatika*, vol. 57, no. 1, pp. 212–220, 2016, ISSN: 0005-1144. DOI: 10.7305/automatika.2016.07.618.

[41] R. Zappulla, J. Virgili-Llop, and M. Romano, "Spacecraft thruster control via sigma–delta modulation," *Journal of Guidance, Control, and Dynamics*, vol. 40, no. 11, pp. 2928–2933, 2017, ISSN: 0731-5090. DOI: 10.2514/1.G002986.

[42] R. L. McKinney, "Positive bases for linear spaces," *Transactions of the American Mathematical Society*, vol. 103, no. 1, p. 131, 1962, ISSN: 0002-9947. DOI: 10.1090/S0002-9947-1962-0147879-X.

[43] R. T. Rockafellar, *Convex analysis*, ser. Princeton paperbacks. Princeton, N.J: Princeton University Press, 1997, ISBN: 0-691-08069-0. DOI: 10.2307/j.ctt14bs1ff.

[44] R. G. Regis, "On the properties of positive spanning sets and positive bases," *Optimization and Engineering*, vol. 17, no. 1, pp. 229–262, 2016, ISSN: 1389-4420. DOI: 10.1007/s11081-015-9286-x.

[45] R. M. Murray, Z. Li, and S. Sastry, *A mathematical introduction to robotic manipulation*. Boca Raton, Fla.: CRC Press, 1994, ISBN: 9780849379819.

[46] J. C. Trinkle, "A quantitative test for form closure grasps," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, IEEE, July 7-10,1992, pp. 1670–1677, ISBN: 0-7803-0738-0. DOI: 10.1109/IROS.1992.594246.

[47] A. Bicchi, "On the force-closure property of robotic grasping," *IFAC Proceedings Volumes*, vol. 27, no. 14, pp. 213–218, 1994, ISSN: 14746670. DOI: 10.1016/S1474-6670(17)47317-4.

[48] D. Prattichizzo and J. C. Trinkle, "Grasping," in *Springer Handbook of Robotics*, B. Siciliano and O. Khatib, Eds., vol. 11, Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 671–700, ISBN: 978-3-540-23957-4. DOI: 10.1007/978-3-540-30301-5{\textunderscore}29.

[49] A. Bicchi, "On the closure properties of robotic grasping," *The International Journal of Robotics Research*, vol. 14, no. 4, pp. 319–334, 1995, ISSN: 0278-3649. DOI: 10.1177/027836499501400402.

[50] B. Mishra, J. T. Schwartz, and M. Sharir, "On the existence and synthesis of multifinger positive grips," *Algorithmica*, vol. 2, no. 1-4, pp. 541–558, 1987, ISSN: 0178-4617. DOI: 10.1007/BF01840373.

[51] Y.-H. Liu, "Qualitative test and force optimization of 3-d frictional form-closure grasps using linear programming," *IEEE Transactions on Robotics and Automation*, vol. 15, no. 1, pp. 163–173, 1999, ISSN: 1042296X. DOI: 10.1109/70.744611.

[52] ——, "Computing n-finger form-closure grasps on polygonal objects," *The International Journal of Robotics Research*, vol. 19, no. 2, pp. 149–158, 2000, ISSN: 0278-3649. DOI: 10.1177/02783640022066798.

[53] B.-H. Kim, S.-R. Oh, B.-J. Yi, and I. H. Suh, "Optimal grasping based on non-dimensionalized performance indices," in *Proceedings 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems. Expanding the Societal Role of Robotics in the the Next Millennium (Cat. No.01CH37180)*, IEEE, 29 Oct.-3 Nov. 2001, pp. 949–956, ISBN: 0-7803-6612-3. DOI: 10.1109/IROS.2001.976291.

[54] H. Dai, A. Majumdar, and R. Tedrake, "Synthesis and optimization of force closure grasps via sequential semidefinite programming," 2015, ISBN: 978-3-319-51531-1.

[55] R. Krug, Y. Bekiroglu, and M. A. Roa, "Grasp quality evaluation done right: How assumed contact force bounds affect wrench-based quality metrics," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 29.05.2017 - 03.06.2017, pp. 1595–1600, ISBN: 978-1-5090-4633-1. DOI: 10.1109/ICRA.2017.7989189.

[56] F. Pierrot, M. Benoit, and P. Dauchez, "Optimal thruster configuration for omnidirectional underwater vehicles. samos: A pythagorean solution," in *IEEE Oceanic Engineering Society. OCEANS'98. Conference Proceedings (Cat. No.98CH36259)*, IEEE, 28 Sept.-1 Oct. 1998, pp. 655–659, ISBN: 0-7803-5045-6. DOI: 10.1109/OCEANS.1998.724320.

[57] C. Chin, M. Shing Lau, E. Low, and G. Lee Seet, "Design of thrusters configuration and thrust allocation control for a remotely operated vehicle," in *2006 IEEE Conference on Robotics, Automation and Mechatronics*, IEEE, 1.06.2006 - 03.06.2006, pp. 1–6, ISBN: 1-4244-0024-4. DOI: 10.1109/RAMECH.2006.252670.

[58] H. Kharrat, "Optimization of thruster configuration for swimming robots," Master's thesis, Rice University, Houston, Texas, 2014.

[59] B. CRAWFORD, "Configuration design and efficient operation of redundant multijet systems," in *AIAA Guidance, Control, and Flight Mechanics Conference*, Reston, Virigina: American Institute of Aeronautics and Astronautics, 8181969. DOI: 10.2514/6.1969-845.

[60] P. A. Servidia and R. S. Sanchez Pena, "Thruster design for position/attitude control of spacecraft," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 38, no. 4, pp. 1172–1180, 2002, ISSN: 0018-9251. DOI: 10.1109/TAES.2002.1145741.

[61] K. M. Lynch, "Controllability of a planar body with unilateral thrusters," *IEEE Transactions on Automatic Control*, vol. 44, no. 6, pp. 1206–1211, 1999, ISSN: 00189286. DOI: 10.1109/9.769375.

[62] M. Wang and Y. Xie, "Control capability analysis for complex spacecraft thruster configurations," *Science China Technological Sciences*, vol. 53, no. 8, pp. 2089–2096, 2010, ISSN: 1674-7321. DOI: 10.1007/s11431-010-3139-9.

[63] P. Wiktor, J.-H. Chen, and D. DeBra, "Optimal thruster configurations for the gp-b spacecraft," in *Automatic Control in Aerospace 1989*, Elsevier, 1990, pp. 203–208, ISBN: 9780080370279. DOI: 10.1016/B978-0-08-037027-9.50032-X.

[64] H.-P. Jin, P. Wiktor, and D. B. DeBra, "An optimal thruster configuration design and evaluation for quick step," *Control Engineering Practice*, vol. 3, no. 8, pp. 1113–1118, 1995, ISSN: 09670661. DOI: 10.1016/0967-0661(95)00104-3.

[65] M. Pasand, A. Hassani, and M. Ghorbani, "A study of spacecraft reaction thruster configurations for attitude control system," *IEEE Aerospace and Electronic Systems Magazine*, vol. 32, no. 7, pp. 22–39, 2017, ISSN: 0885-8985. DOI: 10.1109/MAES.2017.160104.

[66] M. Wang and Y. Xie, "Design of the optimal thruster combinations table for the real time control allocation of spacecraft thrusters," in *Proceedings of the 48h IEEE Conference on Decision and Control (CDC) held jointly with 2009 28th Chinese Control Conference*, IEEE, 15.12.2009 - 18.12.2009, pp. 5063–5068, ISBN: 978-1-4244-3871-6. DOI: 10.1109/CDC.2009.5400664.

[67] H. Unbehauen, *Zustandsregelungen, digitale und nichtlineare Regelsysteme: Mit 9 Tabellen*, 9., durchges. und korrigierte Aufl., korrig. Nachdr, ser. Automatisierungstechnik. Wiesbaden: Vieweg+Teubner, 2009, vol. / Heinz Unbehauen ; 2, ISBN: 978-3-528-83348-0.

[68] A. Braun, *Digitale Regelungstechnik*, Reprint 2018. Berlin and Boston: Oldenbourg Wissenschaftsverlag, 2018, ISBN: 3-486-24027-7. DOI: 10.1515/9783486792324.

[69] J. Lunze, *Regelungstechnik 1: Systemtheoretische Grundlagen, Analyse und Entwurf einschleifiger Regelungen*, 10., aktual. Aufl., ser. Springer-Lehrbuch. Berlin: Springer Vieweg, 2014, ISBN: 978-3-642-53909-1. DOI: 10.1007/978-3-642-53909-1.

[70] S. Zacher and M. Reuter, *Regelungstechnik für Ingenieure: Analyse, Simulation und Entwurf von Regelkreisen*, 14., korrig. Aufl. Wiesbaden: Springer Vieweg, 2014, ISBN: 978-3-8348-2216-1. DOI: 10.1007/978-3-8348-2216-1.

[71] S. W. Jeon and S. Jung, "Hardware-in-the-loop simulation for the reaction control system using pwm-based limit cycle analysis," *IEEE Transactions on Control Systems Technology*, vol. 20, no. 2, pp. 538–545, 2012, ISSN: 1063-6536. DOI: 10.1109/TCST.2011.2117427.

[72] A. M. Giordano, G. Garofalo, and A. Albu-Schaffer, "Momentum dumping for space robots," in *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*, IEEE, 12.12.2017 - 15.12.2017, pp. 5243–5248, ISBN: 978-1-5090-2873-3. DOI: 10.1109/CDC.2017.8264434.

[73] A. M. Giordano, D. Calzolari, and A. Albu-Schaffer, "Workspace fixation for free-floating space robot operations," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 21.05.2018 - 25.05.2018, pp. 889–896, ISBN: 978-1-5386-3081-5. DOI: 10.1109/ICRA.2018.8460478.

[74] A. M. Giordano, C. Ott, and A. Albu-Schaffer, "Coordinated control of spacecraft's attitude and end-effector for space robots," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 2108–2115, 2019. DOI: 10.1109/LRA.2019.2899433.

[75] J. Artigas, M. de Stefano, W. Rackl, R. Lampariello, B. Brunner, W. Bertleff, R. Burger, O. Porges, A. Giordano, C. Borst, and A. Albu-Schaeffer, "The oos-sim: An on-ground simulation facility for on-orbit servicing robotic operations," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 26.05.2015 - 30.05.2015, pp. 2854–2860, ISBN: 978-1-4799-6923-4. DOI: 10.1109/ICRA.2015.7139588.

[76] A. Poo Gallardo, H. Mishra, A. Massimo Giordano, and R. Lampariello, "Robust estimation of motion states for free-floating tumbling target capture," in *2019 IEEE Aerospace Conference*, IEEE, 2.03.2019 - 09.03.2019, pp. 1–11, ISBN: 978-1-5386-6854-2. DOI: 10.1109/AERO.2019.8741802.

[77] H. Mishra, M. de Stefano, A. M. Giordano, and C. Ott, "A nonlinear observer for free-floating target motion using only pose measurements," in *2019 American Control Conference (ACC)*, IEEE, 10.07.2019 - 12.07.2019, pp. 1114–1121, ISBN: 978-1-5386-7926-5. DOI: 10.23919/ACC.2019.8814815.

[78] T. Frankel, *The geometry of physics: An introduction*, 3rd ed. Cambridge and New York: Cambridge University Press, 2012, ISBN: 978-1-107-60260-1.

[79] P. J. Olver, *Applications of Lie groups to differential equations*, 2. ed, corr. 3rd pr, ser. Graduate texts in mathematics. New York: Springer, 1998, vol. 107, ISBN: 9780387940076.

A

Wrench Mapping Between Frames

In this chapter we will derive in more depth, how the adjoint transformation matrix $\boldsymbol{Ad}_A^B$ is generated. It will be used in order to map wrenches from one frame A to another frame B. The matrix will be constructed for the transformation of twists $\boldsymbol{\xi} = [\boldsymbol{v} \;\; \boldsymbol{\omega}] \in \mathbb{R}^6$. $\boldsymbol{v}$ is the translational velocity and $\boldsymbol{\omega}$ is the angular velocity of a body. By using the notion of duality from differential geometry, this transformation can be also applied to wrenches.

This chapter sustains on the even more elaborate derivations which can be found in [45] (Chapter 2 and Appendix A).

A.1 Twist Mapping

One can in general map the translational and angular velocities of a rigid body from a frame A to another frame B, if the homogeneous transformation $\boldsymbol{H}_A^B \left(\boldsymbol{R}_A^B, \boldsymbol{p}_A^B\right) \in SE(3)$ is known. Here, it is assumed that $\boldsymbol{H}_A^B$ is constant with respect to time. Note that $\boldsymbol{R}_A^B$ is the rotation matrix which maps from frame A to B. $\boldsymbol{p}_A^B$ is the position of frame A expressed in frame B.

The mapping can be expressed with

$$\begin{aligned} \boldsymbol{\omega}^B &= \boldsymbol{R}_A^B \boldsymbol{\omega}^A \\ \boldsymbol{v}^B &= \boldsymbol{R}_A^B \boldsymbol{v}^A + \boldsymbol{R}_A^B \left(\boldsymbol{\omega}^A \times \boldsymbol{p}_B^A\right) \end{aligned} \tag{A.1}$$

We can simplify the translational velocity term by using

$$\boldsymbol{p}_B^A = -\boldsymbol{R}_B^A \boldsymbol{p}_A^B \tag{A.2}$$

This relation can be easily obtained from the inverted homogenous transformation and coefficient comparison.

Additionally, one can use the property of commutativity between the cross product and the rotation matrix. We have $\boldsymbol{R} \in SO(3)$ and two vectors $\boldsymbol{a}, \boldsymbol{b} \in \mathbb{R}^3$

$$\boldsymbol{R}\left(\boldsymbol{a} \times \boldsymbol{b}\right) = \left(\boldsymbol{R}\boldsymbol{a}\right) \times \left(\boldsymbol{R}\boldsymbol{b}\right) \tag{A.3}$$

and also the property of anti-commutativity of the cross product with respect to its arguments

$$\boldsymbol{a} \times \boldsymbol{b} = -\boldsymbol{b} \times \boldsymbol{a} \tag{A.4}$$

We can also introduce $\hat{a} \in \mathbb{R}^{3x3}$ as a linear operator for $\boldsymbol{b}$ to represent the cross product.

$$\boldsymbol{a} \times \boldsymbol{b} = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} = \hat{\boldsymbol{a}}\boldsymbol{b} \tag{A.5}$$

These four properties can be used in order to reformulate the translational velocity mapping.

$$\begin{aligned} \boldsymbol{v}^B = \boldsymbol{R}_A^B \boldsymbol{v}^A + \boldsymbol{R}_A^B \left(\boldsymbol{\omega}^A \times \boldsymbol{p}_B^A \right) = \boldsymbol{R}_A^B \boldsymbol{v}^A + \left(\boldsymbol{R}_A^B \boldsymbol{\omega}^A \right) \times \left(\boldsymbol{R}_A^B \boldsymbol{p}_B^A \right) \\ = \boldsymbol{R}_A^B \boldsymbol{v}^A + \left(\boldsymbol{R}_A^B \boldsymbol{\omega}^A \right) \times \left(-\boldsymbol{R}_A^B \boldsymbol{R}_B^A \boldsymbol{p}_A^B \right) = \boldsymbol{R}_A^B \boldsymbol{v}^A + \left(\boldsymbol{R}_A^B \boldsymbol{\omega}^A \right) \times \left(-\boldsymbol{p}_A^B \right) \\ = \boldsymbol{R}_A^B \boldsymbol{v}^A + \hat{\boldsymbol{p}}_A^B \boldsymbol{R}_A^B \boldsymbol{\omega}^A \end{aligned} \tag{A.6}$$

Now we are finally able to construct the adjoint transformation matrix

$$\boldsymbol{\xi}^B = \begin{bmatrix} \boldsymbol{v}^B \\ \boldsymbol{\omega}^B \end{bmatrix} = \begin{bmatrix} \boldsymbol{R}_A^B & \hat{\boldsymbol{p}}_A^B \boldsymbol{R}_A^B \\ \boldsymbol{0} & \boldsymbol{R}_A^B \end{bmatrix} \begin{bmatrix} \boldsymbol{v}^A \\ \boldsymbol{\omega}^A \end{bmatrix} = \boldsymbol{A}\boldsymbol{d}_A^B \begin{bmatrix} \boldsymbol{v}^A \\ \boldsymbol{\omega}^A \end{bmatrix} \tag{A.7}$$

A.2 Power Pairs

We can use the notion of duality, which is a concept from differential geometry, to transfer the adjoint transformation matrix to the use-case of wrench mapping. From a differential geometry point of view, we can observe the twist as a contravariant vector (or also called vector) and the Cartesian wrench as a covector. It is important to distinguish vectors and co-vectors, as they transform in a different way between coordinate systems. This notion is for example used in order to map from joint to Cartesian space with the help of the Jacobian matrix.

Also, the product between a vector and a co-vector is well defined and is invariant with respect to a change in representation. The result will always be a scalar and is independent from the coordinate system. In our case it will be the power P.

Once again we have the frames A and B and the respective twists and wrenches expressed in a specific frame indicated by a right superscript.

$$\left(\boldsymbol{T}^B \right)^T \boldsymbol{\xi}^B = P^B = P^A = \left(\boldsymbol{T}^A \right)^T \boldsymbol{\xi}^A \tag{A.8}$$

A further, more formal insight into differential geometry would be beyond the scope of this work but can be found for example in [78] or [79].

A.3 Wrench Mapping

By combining the results of equation A.7 and A.8, we can relate the wrenches of two frames A and B

$$\begin{aligned} \left(\boldsymbol{F}^B\right)^T \boldsymbol{\xi}^B &= \left(\boldsymbol{F}^A\right)^T \boldsymbol{\xi}^A \\ \left(\boldsymbol{F}^B\right)^T \boldsymbol{Ad}_A^B \, \boldsymbol{\xi}^A &= \left(\boldsymbol{F}^A\right)^T \boldsymbol{\xi}^A \end{aligned} \tag{A.9}$$

After taking the transpose on both sides, we finally have the desired wrench mapping

$$\boldsymbol{T}^A = \begin{bmatrix} \boldsymbol{F}^A \\ \boldsymbol{\tau}^A \end{bmatrix} = \left(\boldsymbol{Ad}_A^B\right)^T \boldsymbol{T}^B = \begin{bmatrix} \boldsymbol{R}_B^A & \boldsymbol{0} \\ -\boldsymbol{R}_B^A \hat{\boldsymbol{p}}_A^B & \boldsymbol{R}_B^A \end{bmatrix} \begin{bmatrix} \boldsymbol{F}^B \\ \boldsymbol{\tau}^B \end{bmatrix} \tag{A.10}$$

The equation uses the knowledge that $\left(\boldsymbol{R}_A^B\right)^T = \boldsymbol{R}_B^A$. Note that we could still reformulate the solution slightly by using equation A.2.

B

Additional Tables for Static Thruster Actuation Comparison

For completeness, the full set of missing tables, which were obtained by the static input-output point of view from Section 2.5, are given in the following. The plotted parameters are introduced in Subsection 2.5.3 and will not be explained here in detail.

B.1 Missing Tables for First Scenario

The first scenario is presented in Subsection 2.5.3. In total a number of 150,000 wrenches are drawn. Each part of the wrench is independently drawn from the force $[-40N, ..., 40N]$ and the torque set $[-40Nm, ..., 40Nm]$.

$f_{ctrl}[Hz]$	**1.11**		**3.33**		**6.67**		**13.33**	
$f_{pwm}[Hz]$	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**
LP1	0.008	0.008	0.072	0.072	0.270	0.270	0.985	0.985
LP2	0.004	0.004	0.036	0.036	0.135	0.135	0.492	0.492
LP3	0.000	0.000	0.003	0.000	0.012	0.000	0.047	0.000
MILP1	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
MILP2	0.031	0.000	0.093	0.001	0.195	0.002	0.427	0.004
MILP3	0.249	0.249	0.249	0.249	0.250	0.250	0.249	0.249
QP1	0.046	0.046	0.192	0.192	0.550	0.550	1.623	1.623
QP2	0.034	0.034	0.194	0.194	0.567	0.567	1.576	1.576
PI1	3.785	3.785	3.788	3.788	3.808	3.808	4.035	4.035
PI2	2.560	2.560	2.613	2.613	2.784	2.784	3.463	3.463
IPI	0.079	0.079	0.159	0.159	0.420	0.420	1.356	1.356

Table B.1: Scenario 1: Sum of absolute torque errors after assignment [N]

$f_{ctrl}[Hz]$	**1.11**		**3.33**		**6.67**		**13.33**	
$f_{pwm}[Hz]$	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**
LP1	0.052	0.009	0.188	0.073	0.463	0.272	1.244	0.987
LP2	0.048	0.005	0.155	0.037	0.337	0.137	0.780	0.494
LP3	0.047	0.000	0.140	0.001	0.279	0.003	0.558	0.006
MILP1	0.047	0.000	0.143	0.001	0.290	0.003	0.593	0.006
MILP2	0.060	0.001	0.181	0.002	0.366	0.004	0.745	0.007
MILP3	0.282	0.249	0.352	0.251	0.471	0.252	0.756	0.254
QP1	0.095	0.046	0.314	0.193	0.741	0.551	1.860	1.624
QP2	0.085	0.035	0.313	0.195	0.746	0.568	1.802	1.577
PI1	3.772	3.785	3.743	3.787	3.709	3.807	3.811	4.032
PI2	2.608	2.560	2.744	2.615	3.011	2.786	3.825	3.466
IPI	0.136	0.080	0.304	0.160	0.652	0.422	1.669	1.359

Table B.2: Scenario 1: Sum of absolute torque errors after modulation [N]

B.2 Tables for Second Scenario

The discussion regarding the second scenario can be found in Subsection 2.5.4. 6,000 wrench vectors with small lengths are sampled from the limited force $[-1N, ..., 1N]$ and the torque set $[-1Nm, ..., 1Nm]$.

Absolute Wrench error

Due to the uniformly distributed sampling, a force in one direction will have, on average, an absolute value of $0.5N$. The sum of the absolute force errors will thus have a maximum of around $1.5N$ if no thruster is actuated at all.

$f_{ctrl}[Hz]$	**1.11**		**3.33**		**6.67**		**13.33**	
$f_{pwm}[Hz]$	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**
LP1	0.134	0.134	0.744	0.744	1.459	1.459	1.492	1.492
LP2	0.068	0.068	0.404	0.404	1.002	1.002	1.522	1.522
LP3	0.006	0.000	0.053	0.000	0.188	0.000	0.563	0.000
MILP1	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
MILP2	0.087	0.001	0.350	0.003	0.865	0.007	1.541	0.012
MILP3	0.201	0.201	0.203	0.203	0.208	0.208	0.208	0.208
QP1	0.297	0.297	1.006	1.006	1.476	1.476	1.492	1.492
QP2	0.284	0.284	1.002	1.002	1.473	1.473	1.492	1.492
PI1	0.371	0.371	1.191	1.191	1.492	1.492	1.492	1.492
PI2	0.302	0.302	1.247	1.247	1.492	1.492	1.492	1.492
IPI	0.302	0.302	1.247	1.247	1.492	1.492	1.492	1.492

Table B.3: Scenario 2: Sum of absolute force errors after assignment [N]

$f_{ctrl}[Hz]$	**1.11**		**3.33**		**6.67**		**13.33**	
$f_{pwm}[Hz]$	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**
LP1	0.168	0.134	0.759	0.744	1.454	1.459	1.492	1.492
LP2	0.101	0.069	0.432	0.404	0.998	1.002	1.522	1.522
LP3	0.067	0.001	0.199	0.002	0.387	0.004	0.713	0.008
MILP1	0.068	0.001	0.144	0.001	0.234	0.003	0.458	0.005
MILP2	0.115	0.001	0.376	0.004	0.856	0.008	1.541	0.016
MILP3	0.246	0.201	0.316	0.205	0.401	0.209	0.580	0.211
QP1	0.314	0.297	1.013	1.006	1.474	1.476	1.492	1.492
QP2	0.301	0.284	1.007	1.002	1.472	1.473	1.492	1.492
PI1	0.317	0.371	1.203	1.191	1.492	1.492	1.492	1.492
PI2	0.317	0.303	1.260	1.247	1.492	1.492	1.492	1.492
IPI	0.317	0.303	1.260	1.247	1.492	1.492	1.492	1.492

Table B.4: Scenario 2: Sum of absolute force errors after modulation [N]

$f_{ctrl}[Hz]$	**1.11**		**3.33**		**6.67**		**13.33**	
$f_{pwm}[Hz]$	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**
LP1	0.122	0.122	0.665	0.665	1.432	1.432	1.495	1.495
LP2	0.058	0.058	0.300	0.300	0.687	0.687	1.426	1.426
LP3	0.006	0.000	0.050	0.000	0.169	0.000	0.497	0.000
MILP1	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
MILP2	0.071	0.001	0.257	0.003	0.602	0.007	1.316	0.016
MILP3	0.198	0.198	0.189	0.189	0.185	0.185	0.201	0.201
QP1	0.219	0.219	0.828	0.828	1.467	1.467	1.495	1.495
QP2	0.211	0.211	0.816	0.816	1.463	1.463	1.495	1.495
PI1	0.312	0.312	1.141	1.141	1.495	1.495	1.495	1.495
PI2	0.223	0.223	1.232	1.232	1.495	1.495	1.495	1.495
IPI	0.223	0.223	1.232	1.232	1.495	1.495	1.495	1.495

Table B.5: Scenario 2: Sum of absolute torque errors after assignment [N]

$f_{ctrl}[Hz]$	**1.11**		**3.33**		**6.67**		**13.33**	
$f_{pwm}[Hz]$	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**
LP1	0.155	0.123	0.701	0.665	1.435	1.432	1.495	1.495
LP2	0.081	0.058	0.314	0.300	0.689	0.687	1.426	1.426
LP3	0.065	0.001	0.192	0.002	0.374	0.004	0.701	0.008
MILP1	0.062	0.001	0.123	0.001	0.188	0.002	0.358	0.004
MILP2	0.091	0.001	0.282	0.004	0.609	0.008	1.317	0.018
MILP3	0.240	0.198	0.279	0.190	0.325	0.186	0.513	0.204
QP1	0.235	0.220	0.866	0.828	1.469	1.467	1.495	1.495
QP2	0.227	0.211	0.853	0.816	1.465	1.463	1.495	1.495
PI1	0.256	0.311	1.162	1.141	1.495	1.495	1.495	1.495
PI2	0.249	0.223	1.250	1.232	1.495	1.495	1.495	1.495
IPI	0.249	0.223	1.250	1.232	1.495	1.495	1.495	1.495

Table B.6: Scenario 2: Sum of absolute torque errors after modulation [N]

$f_{ctrl}[Hz]$	**1.11**		**3.33**		**6.67**		**13.33**	
$f_{pwm}[Hz]$	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**
LP1	28.07	24.26	76.85	77.74	98.58	98.79	100.00	100.00
LP2	18.05	13.82	56.90	55.04	104.95	105.08	101.68	101.68
LP3	10.95	0.16	25.33	0.45	44.99	0.83	71.42	1.56
MILP1	9.52	0.15	19.52	0.40	31.52	0.75	55.97	1.21
MILP2	16.66	0.30	44.76	1.04	76.71	1.96	102.66	3.11
MILP3	35.26	31.97	42.40	34.77	58.95	37.68	60.52	36.89
QP1	35.27	36.45	81.07	81.18	99.34	99.39	100.00	100.00
QP2	62.71	60.32	79.74	79.92	99.25	99.31	100.00	100.00
PI1	34.89	35.57	88.16	87.76	100.00	100.00	100.00	100.00
PI2	35.25	33.93	89.60	89.13	100.00	100.00	100.00	100.00
IPI	35.25	33.93	89.60	89.13	100.00	100.00	100.00	100.00

Table B.7: Scenario 2: Relative force errors after modulation [%]

$f_{ctrl}[Hz]$	**1.11**		**3.33**		**6.67**		**13.33**	
$f_{pwm}[Hz]$	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**
LP1	40.37	37.47	136.12	143.81	99.94	100.07	100.00	100.00
LP2	22.46	18.76	79.21	83.62	201.45	201.44	103.03	103.03
LP3	12.80	0.12	33.28	0.37	58.76	0.80	97.83	1.60
MILP1	7.59	0.09	25.72	0.30	49.08	0.54	93.47	1.14
MILP2	18.81	0.19	63.12	1.11	120.71	2.45	95.37	5.10
MILP3	54.62	52.25	65.20	56.17	82.60	62.40	113.13	60.38
QP1	31.49	31.09	67.94	66.44	98.97	98.93	100.00	100.00
QP2	43.09	44.57	68.97	67.60	98.82	98.77	100.00	100.00
PI1	26.40	28.29	84.65	83.77	100.00	100.00	100.00	100.00
PI2	26.67	25.28	88.52	87.73	100.00	100.00	100.00	100.00
IPI	26.67	25.28	88.52	87.73	100.00	100.00	100.00	100.00

Table B.8: Scenario 2: Relative torque errors after modulation [%]

$f_{ctrl}[Hz]$	**1.11**		**3.33**		**6.67**		**13.33**	
$f_{pwm}[Hz]$	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**
LP1	0.091	0.096	0.351	0.360	0.497	0.494	0.491	0.491
LP2	0.045	0.048	0.177	0.176	0.399	0.397	0.509	0.509
LP3	0.027	0.000	0.081	0.001	0.159	0.002	0.293	0.003
MILP1	0.027	0.000	0.060	0.001	0.097	0.001	0.195	0.002
MILP2	0.050	0.001	0.150	0.001	0.317	0.003	0.545	0.006
MILP3	0.081	0.073	0.103	0.071	0.134	0.071	0.216	0.072
QP1	0.172	0.169	0.501	0.500	0.500	0.499	0.491	0.491
QP2	0.162	0.159	0.497	0.496	0.501	0.499	0.491	0.491
PI1	0.160	0.192	0.517	0.525	0.491	0.491	0.491	0.491
PI2	0.149	0.146	0.494	0.501	0.491	0.491	0.491	0.491
IPI	0.149	0.146	0.494	0.501	0.491	0.491	0.491	0.491

Table B.9: Scenario 2: Standard deviation of sum of absolute force errors after modulation [N]

Amount of Corrected Thrust on Average after Assignment Step

$f_{ctrl}[Hz]$	**1.11**		**3.33**		**6.67**		**13.33**	
$f_{pwm}[Hz]$	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**
LP1	0.145	0.145	0.952	0.952	2.034	2.034	2.111	2.111
LP2	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
LP3	0.006	0.000	0.056	0.000	0.207	0.000	0.687	0.000
MILP1	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
MILP2	0.109	0.001	0.462	0.004	1.211	0.010	2.398	0.022
MILP3	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
QP1	0.384	0.384	1.380	1.380	2.054	2.054	2.082	2.082
QP2	0.379	0.379	1.388	1.388	2.076	2.076	2.108	2.108
PI1	0.688	0.688	5.145	5.145	5.530	5.530	5.530	5.530
PI2	0.466	0.466	2.160	2.160	2.426	2.426	2.426	2.426
IPI	0.466	0.466	2.160	2.160	2.426	2.426	2.426	2.426

Table B.10: Scenario 2: Amount of corrected thrust after the assignment [N]

Average Overactuation

$f_{ctrl}[Hz]$	**1.11**		**3.33**		**6.67**		**13.33**	
$f_{pwm}[Hz]$	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**
LP1	0.041	0.046	0.121	0.149	0.017	0.022	0.000	0.000
LP2	0.029	0.035	0.192	0.212	0.474	0.478	0.085	0.085
LP3	0.007	0.000	0.021	0.000	0.041	0.000	0.069	0.001
MILP1	0.006	0.000	0.021	0.000	0.065	0.001	0.130	0.001
MILP2	0.021	0.000	0.111	0.001	0.371	0.002	0.185	0.005
MILP3	0.008	0.006	0.013	0.004	0.030	0.006	0.066	0.007
QP1	0.056	0.074	0.068	0.087	0.004	0.006	0.000	0.000
QP2	0.057	0.075	0.072	0.092	0.005	0.006	0.000	0.000
PI1	0.248	0.312	0.026	0.032	0.000	0.000	0.000	0.000
PI2	0.068	0.093	0.008	0.011	0.000	0.000	0.000	0.000
IPI2	0.068	0.093	0.008	0.011	0.000	0.000	0.000	0.000

Table B.11: Scenario 2: Average sum of overacted forces after the modulation [N]

$f_{ctrl}[Hz]$	**1.11**		**3.33**		**6.67**		**13.33**	
$f_{pwm}[Hz]$	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**
LP1	0.029	0.035	0.080	0.098	0.006	0.007	0.000	0.000
LP2	0.019	0.030	0.136	0.162	0.295	0.297	0.036	0.036
LP3	0.004	0.000	0.013	0.000	0.026	0.000	0.049	0.000
MILP1	0.003	0.000	0.012	0.000	0.045	0.000	0.092	0.001
MILP2	0.019	0.000	0.083	0.001	0.243	0.003	0.066	0.007
MILP3	0.012	0.012	0.018	0.012	0.042	0.013	0.070	0.013
QP1	0.048	0.067	0.012	0.017	0.000	0.000	0.000	0.000
QP2	0.050	0.070	0.014	0.020	0.000	0.000	0.000	0.000
PI1	0.215	0.277	0.008	0.010	0.000	0.000	0.000	0.000
PI2	0.036	0.055	0.002	0.002	0.000	0.000	0.000	0.000
IPI2	0.036	0.055	0.002	0.002	0.000	0.000	0.000	0.000

Table B.12: Scenario 2: Average sum of overacted torques after the modulation step [Nm]

Average Fuel Consumption

$f_{ctrl}[Hz]$	**1.11**		**3.33**		**6.67**		**13.33**	
$f_{pwm}[Hz]$	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**
LP1	1.90	1.97	1.08	1.16	0.07	0.08	0.00	0.00
LP2	2.04	2.11	1.97	2.04	1.64	1.64	0.14	0.14
LP3	2.48	2.71	5.05	6.46	9.12	13.92	9.98	25.90
MILP1	2.23	2.33	4.27	4.54	8.57	9.10	16.94	17.98
MILP2	1.95	2.32	1.79	4.48	1.78	8.98	0.40	17.30
MILP3	1.79	1.86	3.32	3.50	7.61	8.00	14.17	15.01
QP1	1.63	1.70	0.65	0.70	0.03	0.03	0.00	0.00
QP2	1.66	1.73	0.67	0.72	0.03	0.03	0.00	0.00
PI1	4.58	4.84	0.36	0.39	0.00	0.00	0.00	0.00
PI2	1.86	1.96	0.25	0.27	0.00	0.00	0.00	0.00
IPI	1.86	1.96	0.25	0.27	0.00	0.00	0.00	0.00

Table B.13: Scenario 2: Average fuel consumption [N] (assumed proportional to sum of assignments)

Average Number of Used Thrusters and their Average On-Thrust

$f_{ctrl}[Hz]$	**1.11**		**3.33**		**6.67**		**13.33**	
$f_{pwm}[Hz]$	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**
LP1	4.19	4.19	1.64	1.64	0.07	0.07	0.00	0.00
LP2	5.05	5.05	3.41	3.41	1.64	1.64	0.07	0.07
LP3	7.03	7.80	8.10	10.34	7.64	11.88	4.46	11.90
MILP1	6.63	6.63	6.86	6.86	7.11	7.11	7.21	7.21
MILP2	4.48	6.56	2.95	6.75	1.76	7.01	0.19	6.92
MILP3	4.58	4.58	5.43	5.43	6.31	6.31	6.20	6.20
QP1	4.38	4.38	1.05	1.05	0.03	0.03	0.00	0.00
QP2	4.46	4.46	1.06	1.06	0.03	0.03	0.00	0.00
PI1	15.96	15.96	0.69	0.69	0.00	0.00	0.00	0.00
PI2	6.24	6.24	0.47	0.47	0.00	0.00	0.00	0.00
IPI	6.24	6.24	0.47	0.47	0.00	0.00	0.00	0.00

Table B.14: Scenario 2: Average number of assigned thrusters

$f_{ctrl}[Hz]$	**1.11**		**3.33**		**6.67**		**13.33**	
$f_{pwm}[Hz]$	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**
LP1	0.46	0.48	0.61	0.66	0.07	0.08	0.00	0.00
LP2	0.41	0.42	0.58	0.61	0.92	0.92	0.14	0.14
LP3	0.36	0.35	0.63	0.62	1.20	1.17	2.19	2.18
MILP1	0.34	0.35	0.62	0.66	1.21	1.28	2.35	2.50
MILP2	0.44	0.36	0.61	0.67	0.98	1.28	0.34	2.50
MILP3	0.40	0.41	0.62	0.65	1.21	1.27	2.29	2.42
QP1	0.39	0.40	0.44	0.48	0.03	0.03	0.00	0.00
QP2	0.39	0.41	0.45	0.49	0.03	0.03	0.00	0.00
PI1	0.28	0.30	0.18	0.19	0.00	0.00	0.00	0.00
PI2	0.29	0.31	0.18	0.20	0.00	0.00	0.00	0.00
IPI2	0.29	0.31	0.18	0.20	0.00	0.00	0.00	0.00

Table B.15: Scenario 2: Average assigned thrust of actuated thrusters

Average Computation Time

$f_{ctrl}[Hz]$	**1.11**		**3.33**		**6.67**		**13.33**	
$f_{pwm}[Hz]$	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**
LP1	9.42	8.63	8.63	8.64	8.65	8.73	8.74	9.53
LP2	10.89	8.70	8.73	8.74	8.73	8.74	13.60	8.74
LP3	8.75	8.59	8.56	8.55	8.56	8.57	8.59	8.74
MILP1	71.00	70.66	497.88	498.20	1046.66	1046.70	1197.75	1197.76
MILP2	28.23	70.14	29.37	462.20	32.89	965.75	25.47	1019.19
MILP3	31.13	31.09	182.18	182.21	610.72	611.24	594.11	594.68
QP1	2.36	2.34	2.35	2.35	2.35	2.35	2.35	2.36
QP2	2.00	2.02	1.98	1.98	1.98	1.99	1.98	1.99
PI1	0.02	0.01	0.01	0.01	0.02	1.01	0.02	0.02
PI2	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
IPI	0.16	0.16	0.16	0.16	0.16	0.16	0.19	0.19

Table B.16: Scenario 2: Computation time [ms]

B.3 Tables for Third Scenario

The discussion regarding the third scenario can be found in Subsection 2.5.5. Over 160,000 samples are generated, which are uniformly distributed on a force sphere with radius 10N. This means the euclidean length of the force vector is always equal to $10N$. The torques in the control wrench are set to zero.

Absolute Wrench error

$f_{ctrl}[Hz]$	**1.11**		**3.33**		**6.67**		**13.33**	
$f_{pwm}[Hz]$	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**
LP1	0.017	0.017	0.152	0.152	0.597	0.597	2.255	2.255
LP2	0.009	0.009	0.075	0.075	0.289	0.289	1.050	1.050
LP3	0.001	0.000	0.006	0.000	0.025	0.000	0.098	0.000
MILP1	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
MILP2	0.072	0.001	0.260	0.003	0.644	0.004	1.242	0.008
MILP3	0.297	0.297	0.296	0.296	0.293	0.293	0.280	0.280
QP1	0.099	0.099	0.634	0.634	2.391	2.391	9.624	9.624
QP2	0.069	0.069	0.609	0.609	2.332	2.332	10.204	10.204
PI1	0.079	0.079	0.787	0.787	3.541	3.541	9.609	9.609
PI2	0.068	0.068	0.599	0.599	2.401	2.401	9.609	9.609
IPI	0.068	0.068	0.599	0.599	2.401	2.401	9.609	9.609

Table B.17: Scenario 3: Average sum of absolute force errors after assignment [N]

$f_{ctrl}[Hz]$	**1.11**		**3.33**		**6.67**		**13.33**	
$f_{pwm}[Hz]$	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**
LP1	0.108	0.018	0.403	0.155	1.038	0.602	2.911	2.261
LP2	0.097	0.009	0.303	0.078	0.646	0.292	1.464	1.054
LP3	0.094	0.001	0.281	0.003	0.561	0.006	1.116	0.011
MILP1	0.083	0.001	0.248	0.002	0.488	0.005	0.905	0.009
MILP2	0.153	0.001	0.488	0.005	1.057	0.009	1.865	0.017
MILP3	0.377	0.298	0.535	0.298	0.763	0.298	1.154	0.289
QP1	0.285	0.101	1.106	0.638	3.099	2.398	10.000	9.629
QP2	0.256	0.071	1.083	0.613	3.031	2.340	10.519	10.208
PI1	0.210	0.080	1.027	0.789	3.425	3.539	10.018	9.614
PI2	0.254	0.069	1.080	0.604	3.096	2.408	10.018	9.614
IPI	0.254	0.069	1.080	0.604	3.096	2.408	10.018	9.614

Table B.18: Scenario 3: Average sum of absolute force errors after modulation [N]

$f_{ctrl}[Hz]$	**1.11**		**3.33**		**6.67**		**13.33**	
$f_{pwm}[Hz]$	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**
LP1	0.013	0.013	0.109	0.109	0.407	0.407	1.355	1.355
LP2	0.006	0.006	0.055	0.055	0.204	0.204	0.679	0.679
LP3	0.000	0.000	0.005	0.000	0.018	0.000	0.071	0.000
MILP1	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
MILP2	0.021	0.000	0.080	0.001	0.195	0.002	0.453	0.003
MILP3	0.202	0.202	0.206	0.206	0.207	0.207	0.209	0.209
QP1	0.008	0.008	0.095	0.095	0.193	0.193	0.142	0.142
QP2	0.012	0.012	0.106	0.106	0.318	0.318	0.968	0.968
PI1	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
PI2	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
IPI	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000

Table B.19: Scenario 3: Average sum of absolute torque errors after assignment [N]

$f_{ctrl}[Hz]$	**1.11**		**3.33**		**6.67**		**13.33**	
$f_{pwm}[Hz]$	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**
LP1	0.035	0.013	0.159	0.109	0.466	0.407	1.347	1.355
LP2	0.029	0.007	0.105	0.055	0.267	0.204	0.721	0.680
LP3	0.025	0.000	0.074	0.001	0.147	0.001	0.289	0.003
MILP1	0.022	0.000	0.065	0.001	0.132	0.001	0.292	0.003
MILP2	0.033	0.000	0.106	0.001	0.226	0.002	0.471	0.005
MILP3	0.212	0.202	0.233	0.206	0.253	0.207	0.323	0.209
QP1	0.030	0.008	0.119	0.096	0.199	0.194	0.162	0.142
QP2	0.036	0.012	0.152	0.107	0.361	0.318	0.891	0.967
PI1	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
PI2	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
IPI	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000

Table B.20: Scenario 3: Average sum of absolute torque errors after modulation [N]

$f_{ctrl}[Hz]$	**1.11**		**3.33**		**6.67**		**13.33**	
$f_{pwm}[Hz]$	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**
LP1	2.58	1.68	7.91	5.91	16.49	13.47	34.18	30.47
LP2	1.96	1.09	5.69	3.86	11.01	8.52	21.16	18.67
LP3	1.34	0.07	3.85	0.10	7.10	0.14	12.97	0.23
MILP1	1.28	0.07	3.61	0.09	6.50	0.13	11.09	0.20
MILP2	3.04	0.08	8.30	0.19	15.40	0.25	25.05	0.45
MILP3	6.01	5.17	Inf	Inf	Inf	Inf	Inf	Inf
QP1	8.67	6.93	23.02	19.92	43.14	39.61	81.27	79.93
QP2	8.22	6.45	22.79	19.68	42.55	39.03	83.10	81.96
PI1	Inf	Inf	Inf	Inf	Inf	Inf	81.35	79.87
PI2	8.30	6.54	22.95	19.81	43.28	39.81	81.35	79.87
IPI	8.30	6.54	22.95	19.81	43.28	39.81	81.35	79.87

Table B.21: Scenario 3: Relative force errors after modulation [%]

$f_{ctrl}[Hz]$	**1.11**		**3.33**		**6.67**		**13.33**	
$f_{pwm}[Hz]$	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**
LP1	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf
LP2	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf
LP3	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf
MILP1	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf
MILP2	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf
MILP3	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf
QP1	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf
QP2	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf
PI1	Inf	Inf	Inf	Inf	Inf	Inf	0.00	0.00
PI2	Inf	Inf	Inf	Inf	Inf	Inf	0.00	0.00
IPI	Inf	Inf	Inf	Inf	Inf	Inf	0.00	0.00

Table B.22: Scenario 3: Relative torque errors after modulation [%]

$f_{ctrl}[Hz]$	**1.11**		**3.33**		**6.67**		**13.33**	
$f_{pwm}[Hz]$	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**
LP1	0.052	0.051	0.229	0.245	0.577	0.623	1.390	1.472
LP2	0.032	0.026	0.111	0.119	0.259	0.290	0.595	0.630
LP3	0.027	0.000	0.082	0.001	0.162	0.002	0.330	0.003
MILP1	0.022	0.000	0.067	0.001	0.135	0.001	0.281	0.003
MILP2	0.054	0.001	0.167	0.002	0.350	0.004	0.651	0.006
MILP3	0.029	0.015	0.074	0.018	0.147	0.024	0.291	0.039
QP1	0.137	0.155	0.607	0.689	1.546	1.699	5.246	5.568
QP2	0.131	0.150	0.592	0.676	1.408	1.556	4.620	4.868
PI1	0.151	0.183	0.729	0.903	1.669	1.856	5.296	5.611
PI2	0.158	0.167	0.699	0.771	1.700	1.823	5.296	5.611
IPI	0.158	0.167	0.699	0.771	1.700	1.823	5.296	5.611

Table B.23: Scenario 3: Standard deviation of sum of absolute force errors after modulation [N]

Amount of Corrected Thrust on Average after Assignment Step

$f_{ctrl}[Hz]$	**1.11**		**3.33**		**6.67**		**13.33**	
$f_{pwm}[Hz]$	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**
LP1	0.017	0.017	0.152	0.152	0.597	0.597	2.255	2.255
LP2	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
LP3	0.001	0.000	0.006	0.000	0.025	0.000	0.098	0.000
MILP1	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
MILP2	0.072	0.001	0.261	0.003	0.653	0.004	1.316	0.008
MILP3	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
QP1	0.069	0.069	0.604	0.604	2.361	2.361	9.594	9.594
QP2	0.066	0.066	0.606	0.606	2.329	2.329	10.201	10.201
PI1	0.079	0.079	0.787	0.787	3.715	3.715	19.538	19.538
PI2	0.068	0.068	0.599	0.599	2.401	2.401	9.609	9.609
IPI	0.068	0.068	0.599	0.599	2.401	2.401	9.609	9.609

Table B.24: Scenario 3: Amount of corrected thrust after the assignment [N]

Average Overactuation

$f_{ctrl}[Hz]$	**1.11**		**3.33**		**6.67**		**13.33**	
$f_{pwm}[Hz]$	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**
LP1	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
LP2	0.003	0.004	0.026	0.038	0.102	0.143	0.387	0.507
LP3	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
MILP1	0.000	0.000	0.000	0.000	0.003	0.000	0.020	0.000
MILP2	0.000	0.000	0.003	0.000	0.015	0.000	0.098	0.000
MILP3	0.000	0.000	0.000	0.000	0.002	0.000	0.019	0.000
QP1	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
QP2	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
PI1	0.102	0.079	0.728	0.786	2.936	3.527	0.000	0.000
PI2	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
IPI	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000

Table B.25: Scenario 3: Average sum of overacted forces after the modulation [N]

$f_{ctrl}[Hz]$	**1.11**		**3.33**		**6.67**		**13.33**	
$f_{pwm}[Hz]$	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**
LP1	0.035	0.013	0.159	0.109	0.466	0.407	1.347	1.355
LP2	0.029	0.007	0.105	0.055	0.267	0.204	0.721	0.680
LP3	0.025	0.000	0.074	0.001	0.147	0.001	0.289	0.003
MILP1	0.022	0.000	0.065	0.001	0.132	0.001	0.292	0.003
MILP2	0.033	0.000	0.106	0.001	0.226	0.002	0.471	0.005
MILP3	0.212	0.202	0.233	0.206	0.253	0.207	0.323	0.209
QP1	0.030	0.008	0.119	0.096	0.199	0.194	0.162	0.142
QP2	0.036	0.012	0.152	0.107	0.361	0.318	0.891	0.967
PI1	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
PI2	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
IPI	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000

Table B.26: Scenario 3: Average sum of overacted torques after the modulation [Nm]

Average Fuel Consumption

$f_{ctrl}[Hz]$	**1.11**		**3.33**		**6.67**		**13.33**	
$f_{pwm}[Hz]$	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**
LP1	14.89	14.98	14.60	14.85	13.96	14.40	12.09	12.74
LP2	14.91	15.00	14.75	15.00	14.56	15.00	14.31	14.96
LP3	14.96	15.07	15.21	15.61	16.36	17.40	21.20	24.23
MILP1	14.94	15.02	14.93	15.18	15.36	15.86	18.70	19.67
MILP2	14.85	15.02	14.52	15.17	13.98	15.84	13.33	19.59
MILP3	14.63	14.71	14.60	14.84	14.96	15.43	18.00	18.93
QP1	14.72	14.90	13.90	14.37	11.90	12.61	5.00	5.37
QP2	14.75	14.93	13.92	14.39	11.97	12.66	4.49	4.79
PI1	24.53	24.85	23.30	24.14	19.93	21.20	4.99	5.39
PI2	14.75	14.93	13.92	14.40	11.91	12.60	4.99	5.39
IPI	14.75	14.93	13.92	14.40	11.91	12.60	4.99	5.39

Table B.27: Scenario 3: Average fuel consumption [N] (assumed proportional to sum of assignments)

Average Number of Used Thrusters and their Average On-Thrust

$f_{ctrl}[Hz]$	**1.11**		**3.33**		**6.67**		**13.33**	
$f_{pwm}[Hz]$	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**
LP1	5.46	5.46	5.05	5.05	4.46	4.46	3.35	3.35
LP2	5.56	5.56	5.36	5.36	5.05	5.05	4.46	4.46
LP3	5.79	5.86	6.03	6.26	6.38	6.86	7.00	7.96
MILP1	5.12	5.12	5.36	5.36	5.66	5.66	6.17	6.17
MILP2	4.93	5.12	4.81	5.35	4.63	5.65	4.25	6.18
MILP3	4.93	4.93	5.15	5.15	5.47	5.47	5.98	5.98
QP1	11.17	11.17	9.56	9.56	7.23	7.23	2.39	2.39
QP2	11.20	11.20	9.56	9.56	7.24	7.24	2.07	2.07
PI1	19.07	19.07	16.97	16.97	13.12	13.12	2.40	2.40
PI2	11.20	11.20	9.60	9.60	7.20	7.20	2.40	2.40
IPI	11.20	11.20	9.60	9.60	7.20	7.20	2.40	2.40

Table B.28: Scenario 3: Average number of assigned thrusters

$f_{ctrl}[Hz]$	**1.11**		**3.33**		**6.67**		**13.33**	
$f_{pwm}[Hz]$	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**
LP1	2.77	2.79	2.98	3.03	3.27	3.37	3.75	3.94
LP2	2.71	2.73	2.81	2.86	2.97	3.05	3.33	3.48
LP3	2.62	2.61	2.59	2.54	2.64	2.58	3.09	3.06
MILP1	2.93	2.95	2.81	2.86	2.73	2.82	3.03	3.19
MILP2	3.02	2.95	3.04	2.86	3.06	2.83	3.23	3.18
MILP3	2.98	3.00	2.86	2.91	2.77	2.85	3.02	3.18
QP1	1.34	1.36	1.52	1.57	1.76	1.85	1.28	1.37
QP2	1.34	1.35	1.52	1.57	1.75	1.85	1.44	1.54
PI1	1.29	1.30	1.38	1.43	1.53	1.63	1.25	1.35
PI2	1.34	1.36	1.52	1.57	1.77	1.87	1.25	1.35
IPI	1.34	1.36	1.52	1.57	1.77	1.87	1.25	1.35

Table B.29: Scenario 3: Average assigned thrust of actuated thrusters

Average Computation time

It has to be noted that this scenario was computed on a different machine than the other scenarios. Thus, the run-times are different.

$f_{ctrl}[Hz]$	**1.11**		**3.33**		**6.67**		**13.33**	
$f_{pwm}[Hz]$	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**
LP1	14.40	14.31	14.29	14.30	14.31	14.28	14.32	14.34
LP2	14.43	14.44	14.45	14.45	14.44	14.46	14.48	14.45
LP3	14.25	14.24	14.22	14.25	14.25	14.23	14.28	14.28
MILP1	70.11	69.90	87.49	87.38	136.46	136.03	297.11	296.67
MILP2	67.09	71.07	70.95	89.19	75.94	139.26	84.16	296.69
MILP3	78.54	78.37	89.32	89.29	134.70	134.48	286.65	286.36
QP1	3.64	3.64	3.64	3.64	3.64	3.65	3.66	3.65
QP2	3.55	3.56	3.56	3.55	3.56	3.56	3.57	3.57
PI1	0.02	0.02	0.02	0.02	0.02	0.02	0.02	0.02
PI2	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
IPI	0.22	0.22	0.22	0.22	0.22	0.22	0.22	0.22

Table B.30: Scenario 3: Computation time [ms]

B.4 Tables for Fourth Scenario

The discussion regarding the fourth scenario can be found in Subsection 2.5.6. Over 160,000 samples are generated, which are uniformly distributed on a torque sphere with radius $10Nm$. Thus, this time the torque vector will have a length of $10Nm$. The forces in the control wrench are set to zero.

Absolute Wrench error

$f_{ctrl}[Hz]$	**1.11**		**3.33**		**6.67**		**13.33**	
$f_{pwm}[Hz]$	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**
LP1	0.004	0.004	0.032	0.032	0.121	0.121	0.434	0.434
LP2	0.002	0.002	0.016	0.016	0.061	0.061	0.220	0.220
LP3	0.000	0.000	0.001	0.000	0.005	0.000	0.020	0.000
MILP1	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
MILP2	0.010	0.000	0.033	0.000	0.073	0.001	0.162	0.002
MILP3	0.225	0.225	0.221	0.221	0.219	0.219	0.226	0.226
QP1	0.002	0.002	0.017	0.017	0.048	0.048	0.053	0.053
QP2	0.008	0.008	0.054	0.054	0.145	0.145	0.271	0.271
PI1	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
PI2	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
IPI	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000

Table B.31: Scenario 4: Average sum of absolute force errors after assignment [N]

$f_{ctrl}[Hz]$	**1.11**		**3.33**		**6.67**		**13.33**	
$f_{pwm}[Hz]$	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**
LP1	0.014	0.004	0.057	0.032	0.160	0.122	0.469	0.434
LP2	0.012	0.002	0.042	0.016	0.100	0.061	0.261	0.221
LP3	0.011	0.000	0.032	0.000	0.063	0.001	0.125	0.001
MILP1	0.011	0.000	0.034	0.000	0.072	0.001	0.149	0.002
MILP2	0.014	0.000	0.047	0.000	0.094	0.001	0.201	0.003
MILP3	0.226	0.225	0.220	0.221	0.220	0.218	0.262	0.225
QP1	0.014	0.002	0.047	0.017	0.074	0.048	0.056	0.053
QP2	0.020	0.008	0.078	0.054	0.164	0.145	0.275	0.271
PI1	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
PI2	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
IPI	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000

Table B.32: Scenario 4: Average sum of absolute force errors after modulation [N]

$f_{ctrl}[Hz]$	**1.11**		**3.33**		**6.67**		**13.33**	
$f_{pwm}[Hz]$	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**
LP1	0.014	0.014	0.122	0.122	0.482	0.482	1.809	1.809
LP2	0.007	0.007	0.061	0.061	0.235	0.235	0.859	0.859
LP3	0.000	0.000	0.005	0.000	0.020	0.000	0.078	0.000
MILP1	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
MILP2	0.020	0.000	0.088	0.000	0.271	0.001	0.815	0.004
MILP3	0.247	0.247	0.239	0.239	0.229	0.229	0.228	0.228
QP1	0.043	0.043	0.279	0.279	0.986	0.986	3.059	3.059
QP2	0.034	0.034	0.257	0.257	0.974	0.974	3.040	3.040
PI1	0.060	0.060	0.578	0.578	1.969	1.969	3.222	3.222
PI2	0.046	0.046	0.383	0.383	1.340	1.340	3.692	3.692
IPI	0.046	0.046	0.383	0.383	1.340	1.340	3.692	3.692

Table B.33: Scenario 4: Average sum of absolute torque errors after assignment [N]

$f_{ctrl}[Hz]$	**1.11**		**3.33**		**6.67**		**13.33**	
$f_{pwm}[Hz]$	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**
LP1	0.090	0.015	0.331	0.124	0.849	0.486	2.351	1.814
LP2	0.081	0.008	0.253	0.063	0.539	0.238	1.212	0.862
LP3	0.079	0.001	0.237	0.002	0.472	0.005	0.939	0.010
MILP1	0.078	0.001	0.231	0.002	0.450	0.005	0.865	0.009
MILP2	0.087	0.001	0.268	0.003	0.566	0.005	1.227	0.011
MILP3	0.322	0.248	0.458	0.242	0.659	0.234	1.078	0.236
QP1	0.124	0.044	0.481	0.281	1.288	0.989	3.381	3.062
QP2	0.115	0.035	0.460	0.259	1.274	0.977	3.354	3.043
PI1	0.101	0.061	0.538	0.577	1.538	1.964	2.479	3.213
PI2	0.146	0.047	0.600	0.385	1.624	1.342	4.130	3.695
IPI	0.146	0.047	0.600	0.385	1.624	1.342	4.130	3.695

Table B.34: Scenario 4: Average sum of absolute torque errors after modulation [N]

$f_{ctrl}[Hz]$	**1.11**		**3.33**		**6.67**		**13.33**	
$f_{pwm}[Hz]$	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**
LP1	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf
LP2	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf
LP3	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf
MILP1	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf
MILP2	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf
MILP3	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf
QP1	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf
QP2	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf
PI1	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf
PI2	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf
IPI	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf

Table B.35: Scenario 4: Relative force errors after modulation [%]

$f_{ctrl}[Hz]$	**1.11**		**3.33**		**6.67**		**13.33**	
$f_{pwm}[Hz]$	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**
LP1	1.98	1.23	6.14	4.43	12.92	10.29	27.21	23.84
LP2	1.54	0.80	4.51	2.90	8.87	6.53	17.37	14.75
LP3	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf
MILP1	1.03	0.06	2.80	0.08	5.25	0.11	9.84	0.18
MILP2	1.17	0.07	3.62	0.10	7.20	0.15	15.26	0.27
MILP3	Inf	Inf	Inf	Inf	7.25	Inf	11.51	Inf
QP1	Inf	Inf	4.98	Inf	13.73	Inf	37.89	Inf
QP2	Inf	Inf	4.91	Inf	13.44	Inf	36.66	Inf
PI1	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf
PI2	Inf	Inf	Inf	Inf	Inf	Inf	37.93	Inf
IPI	Inf	Inf	Inf	Inf	Inf	Inf	37.93	36.43

Table B.36: Scenario 4: Relative torque errors after modulation [%]

$f_{ctrl}[Hz]$	**1.11**		**3.33**		**6.67**		**13.33**	
$f_{pwm}[Hz]$	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**
LP1	0.054	0.051	0.233	0.245	0.600	0.644	1.460	1.538
LP2	0.035	0.026	0.116	0.121	0.265	0.305	0.624	0.685
LP3	0.031	0.000	0.094	0.001	0.187	0.002	0.371	0.004
MILP1	0.031	0.000	0.091	0.001	0.175	0.002	0.330	0.003
MILP2	0.031	0.000	0.098	0.001	0.209	0.002	0.469	0.005
MILP3	0.084	0.071	0.133	0.074	0.175	0.077	0.312	0.072
QP1	0.087	0.102	0.380	0.454	0.874	1.028	1.762	1.929
QP2	0.082	0.097	0.350	0.424	0.847	1.003	1.709	1.872
PI1	0.077	0.108	0.356	0.467	0.699	0.774	0.860	1.052
PI2	0.074	0.093	0.318	0.386	0.734	0.811	1.606	1.603
IPI	0.074	0.093	0.318	0.386	0.734	0.811	1.606	1.603

Table B.37: Scenario 4: Standard deviation of sum of absolute torque errors after modulation [N]

Amount of Corrected Thrust on Average after Assignment Step

$f_{ctrl}[Hz]$	**1.11**		**3.33**		**6.67**		**13.33**	
$f_{pwm}[Hz]$	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**
LP1	0.014	0.014	0.122	0.122	0.488	0.488	1.856	1.856
LP2	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
LP3	0.000	0.000	0.005	0.000	0.020	0.000	0.079	0.000
MILP1	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
MILP2	0.020	0.000	0.090	0.000	0.285	0.001	0.896	0.004
MILP3	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
QP1	0.031	0.031	0.271	0.271	1.001	1.001	3.303	3.303
QP2	0.033	0.033	0.259	0.259	1.002	1.002	3.313	3.313
PI1	0.062	0.062	0.631	0.631	3.021	3.021	13.204	13.204
PI2	0.048	0.048	0.427	0.427	1.699	1.699	6.755	6.755
IPI	0.048	0.048	0.427	0.427	1.699	1.699	6.755	6.755

Table B.38: Scenario 4: Amount of corrected thrust after the assignment [N]

Average Overactuation

$f_{ctrl}[Hz]$	**1.11**		**3.33**		**6.67**		**13.33**	
$f_{pwm}[Hz]$	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**
LP1	0.014	0.004	0.057	0.032	0.160	0.122	0.469	0.434
LP2	0.012	0.002	0.042	0.016	0.100	0.061	0.261	0.221
LP3	0.011	0.000	0.032	0.000	0.063	0.001	0.125	0.001
MILP1	0.011	0.000	0.034	0.000	0.072	0.001	0.149	0.002
MILP2	0.014	0.000	0.047	0.000	0.094	0.001	0.201	0.003
MILP3	0.226	0.225	0.220	0.221	0.220	0.218	0.262	0.225
QP1	0.014	0.002	0.047	0.017	0.074	0.048	0.056	0.053
QP2	0.020	0.008	0.078	0.054	0.164	0.145	0.275	0.271
PI1	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
PI2	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
IPI	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000

Table B.39: Scenario 4: Sum of overacted forces after the modulation [N]

$f_{ctrl}[Hz]$	**1.11**		**3.33**		**6.67**		**13.33**	
$f_{pwm}[Hz]$	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**
LP1	0.001	0.001	0.005	0.007	0.017	0.024	0.052	0.078
LP2	0.003	0.004	0.021	0.031	0.080	0.118	0.299	0.418
LP3	0.000	0.000	0.001	0.000	0.003	0.000	0.005	0.000
MILP1	0.000	0.000	0.001	0.000	0.003	0.000	0.010	0.000
MILP2	0.005	0.000	0.017	0.000	0.049	0.000	0.147	0.001
MILP3	0.000	0.000	0.001	0.000	0.002	0.000	0.007	0.000
QP1	0.014	0.012	0.092	0.102	0.293	0.349	0.672	0.851
QP2	0.015	0.013	0.087	0.097	0.290	0.348	0.657	0.840
PI1	0.056	0.058	0.408	0.525	1.313	1.795	1.535	2.446
PI2	0.022	0.023	0.140	0.178	0.425	0.563	0.663	0.894
IPI	0.022	0.023	0.140	0.178	0.425	0.563	0.663	0.894

Table B.40: Scenario 4: Sum of overacted torques after the modulation step [Nm]

Average Fuel Consumption

$f_{ctrl}[Hz]$	**1.11**		**3.33**		**6.67**		**13.33**	
$f_{pwm}[Hz]$	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**
LP1	17.23	17.32	16.97	17.20	16.41	16.84	14.79	15.47
LP2	17.24	17.33	17.09	17.33	16.90	17.33	16.63	17.30
LP3	17.29	17.38	17.45	17.81	18.37	19.27	22.28	24.86
MILP1	17.26	17.34	17.20	17.46	17.40	17.91	19.15	20.16
MILP2	17.23	17.34	17.01	17.46	16.68	17.90	16.07	20.09
MILP3	17.01	17.10	16.93	17.17	17.05	17.54	18.55	19.53
QP1	17.16	17.28	16.73	17.04	15.80	16.30	13.30	14.00
QP2	17.18	17.29	16.76	17.07	15.81	16.32	13.30	14.01
PI1	38.27	38.62	37.08	38.04	34.06	35.64	23.69	25.46
PI2	20.97	21.16	20.27	20.77	18.66	19.50	13.44	14.44
IPI	20.97	21.16	20.27	20.77	18.66	19.50	13.44	14.44

Table B.41: Scenario 4: Average fuel consumption [N] (assumed proportional to sum of assignments)

Average Number of Used Thrusters and their Average On-Thrust

$f_{ctrl}[Hz]$	**1.11**		**3.33**		**6.67**		**13.33**	
$f_{pwm}[Hz]$	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**
LP1	5.18	5.18	4.86	4.86	4.37	4.37	3.45	3.45
LP2	5.27	5.27	5.10	5.10	4.86	4.86	4.37	4.37
LP3	5.45	5.50	5.64	5.82	5.94	6.31	6.46	7.22
MILP1	5.43	5.43	5.63	5.63	5.92	5.92	6.39	6.39
MILP2	5.29	5.43	5.16	5.62	4.98	5.91	4.52	6.36
MILP3	5.21	5.21	5.36	5.36	5.67	5.67	6.26	6.26
QP1	6.91	6.91	6.19	6.19	5.21	5.21	3.64	3.64
QP2	6.90	6.90	6.21	6.21	5.21	5.21	3.64	3.64
PI1	21.27	21.27	19.58	19.58	16.45	16.45	9.47	9.47
PI2	11.43	11.43	10.29	10.29	8.59	8.59	5.22	5.22
IPI	11.43	11.43	10.29	10.29	8.59	8.59	5.22	5.22

Table B.42: Scenario 4: Number of assigned thrusters

$f_{ctrl}[Hz]$	**1.11**		**3.33**		**6.67**		**13.33**	
$f_{pwm}[Hz]$	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**
LP1	3.40	3.42	3.65	3.70	4.03	4.13	4.67	4.87
LP2	3.33	3.35	3.45	3.49	3.64	3.73	4.06	4.22
LP3	3.23	3.23	3.20	3.14	3.22	3.13	3.58	3.47
MILP1	3.23	3.24	3.12	3.16	3.00	3.09	3.07	3.23
MILP2	3.31	3.25	3.39	3.17	3.50	3.10	3.84	3.24
MILP3	3.34	3.36	3.25	3.29	3.11	3.20	3.06	3.22
QP1	2.59	2.60	2.91	2.96	3.35	3.45	4.01	4.21
QP2	2.59	2.60	2.90	2.95	3.35	3.44	4.02	4.21
PI1	1.80	1.82	1.90	1.95	2.08	2.17	2.52	2.71
PI2	1.85	1.86	2.00	2.05	2.22	2.32	2.61	2.81
IPI	1.85	1.86	2.00	2.05	2.22	2.32	2.61	2.81

Table B.43: Scenario 4: Assigned thrust of actuated thrusters

Average Computation Time

$f_{ctrl}[Hz]$	**1.11**		**3.33**		**6.67**		**13.33**	
$f_{pwm}[Hz]$	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**	**333**	**33.3k**
LP1	8.73	8.66	8.65	8.65	8.66	8.66	8.66	8.66
LP2	8.75	8.74	8.74	8.74	8.76	8.75	8.75	8.75
LP3	8.61	8.59	8.58	8.58	8.58	8.58	8.59	8.59
MILP1	34.84	34.81	38.67	38.67	49.53	49.52	97.92	97.90
MILP2	34.47	35.14	35.15	38.95	41.80	50.33	55.55	99.20
MILP3	40.65	40.61	40.99	40.99	50.56	50.56	84.93	84.90
QP1	2.51	2.50	2.51	2.50	2.51	2.54	2.51	2.51
QP2	2.13	2.14	2.13	2.13	2.20	2.13	2.13	2.13
PI1	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
PI2	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
IPI	0.16	0.16	0.16	0.16	0.16	0.16	0.16	0.16

Table B.44: Scenario 4: Computation time [ms]